On Indecomposable Decompositions of Persistence Modules on Commutative Ladders of Finite Type and Persistent Homology of Sampled Maps

博士論文

On Indecomposable Decompositions of Persistence Modules on Commutative Ladders of Finite Type and Persistent Homology of Sampled Maps

[有限型の可換梯子型
パーシステント加群の直既約分解と
サンプル写像のパーシステントホモロジーについて]

竹内　博志

平成３０年

A doctoral thesis submitted to

Tohoku University

On Indecomposable Decompositions of Persistence Modules on Commutative Ladders of Finite Type and Persistent Homology of Sampled Maps

Hiroshi Takeuchi

February 2019

Abstract

Topological data analysis has been developed as a tool of data analysis which can quantitatively capture topological shapes of geometric data (e.g. topological spaces, point clouds). A central role in topological data analysis is played by persistent homology groups, which are sequences of linear maps between homology groups with a field coefficient K. By formulating the persistent homology groups as representations of quivers, the concept of persistent homology groups has been extended to general persistence modules. This extension has enabled to capture topological features of geometric data which the conventional concept of persistent homology cannot deal with.

An example of the extension is persistence modules on commutative ladders, which can simultaneously capture common topological shapes and their robustness of time series geometric data. In general, when we consider real data analysis using persistence modules, it is routine to decompose the persistence modules and extract the generators of homology groups. A general decomposition algorithm using idempotents in the endomorphism rings of the persistence modules is known. While this algorithm can deal with arbitrary types of persistence modules, the computational complexity is too high for the applications.

In order to address this issue, we provide computationally lighter decomposition algorithms specialized for the persistence modules on the commutative ladders. Every persistence module on a commutative ladder can be written in block matrix form by regarding it as a morphism between representations of an A_n type quiver. Our algorithm transforms the block matrix to a normal form by using isomorphic transformations. We prove that the procedure of transformations to the normal form is equivalent to the indecomposable decomposition of the persistence module on the commutative ladder.

As another example of persistence modules available for time series geometric data, we consider persistent homology of sampled maps. A sampled map is a restriction of a continuous map to a finite subset. This can be seen as a map between point clouds, in other words, a movement of point clouds. From this sampled map, we can generate a filtration of simplicial maps. Applying a homology functor to this filtration yields the filtration of homology induced maps, which can capture robust topological shapes of the continuous map. The block matrix form mentioned above enables to define the persistent homology of the filtration of homology induced maps. Moreover, this method generating the persistent homology can be utilized to analyze 2-D persistence modules.

We can prove a stability theorem, that is, 1-Lipschitz continuity of the persistent homology to the sampled map. Thanks to this stability theorem, errors of output (the persistent homology) are bounded by errors of input (the sampled map).

Contents

Chapter 1.

Introduction

In this thesis, we present decomposition methods for broadly two types of persistent homology groups. Persistent homology groups (abbrev. persistent homology), first proposed in the paper [15], are protagonists in the field of topological data analysis, where we analyze robust topological structures (e.g. connected components, holes, cavities, or higher dimensional homology groups) latent in data. Persistent homology has been applied to a wide range of data analyses, for example, cavities in granular materials [26], protein compressibility [19], hierarchical ring structures in amorphous solid [22], and so on.

To start, we will present a small example of topological data analysis using persistent homology. A finite subset of a Euclidean space is called a *point cloud*, which is a typical model of experimental data, such as atomic configurations.

Figure 1.1 is a small example of point cloud P embedded in the 2-dimensional Euclidean space $\mathbb{R}^2$. Ordinarily, 1-dimensional homology groups are used to extract hole

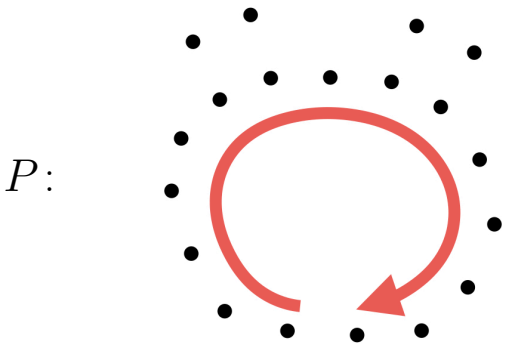
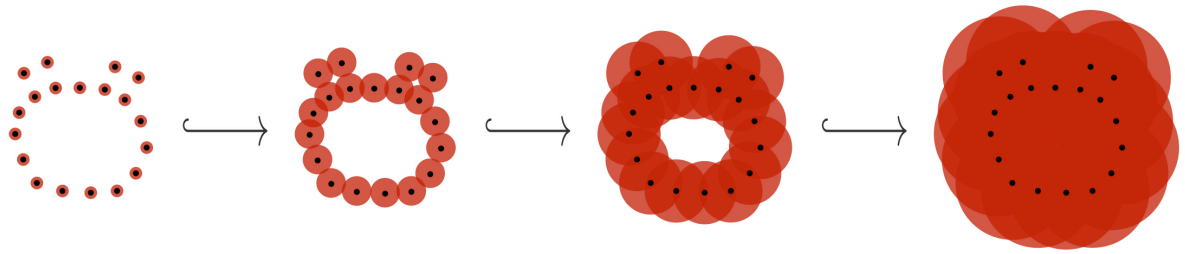


Figure 1.1.: The set P of the black points above is a small example of a point cloud in 2-dimensional Euclidean space. We can observe a hole-like structure illustrated with the red arrow. The machinery of classical homology cannot retrieve such structure since the topology of P is discrete.

structures in topological spaces. Although one natural topology for such a point cloud is the relative topology induced by $\mathbb{R}^2$, since the relative topology is discrete, we cannot extract the "visible" hole with homology. In topological data analysis, such a hole can be formed in the following way.

Consider the filtration $X_1 \subset X_2 \subset X_3 \subset X_4$ in Figure 1.2. Each topological space is



X_1 with radius r_1 | X_2 with radius r_2 | X_3 with radius r_3 | X_4 with radius r_4

Figure 1.2.: A filtration with the four topological spaces X_1, X_2, X_3, and X_4. Each topological space X_i is constructed by expanding the point cloud P with disks of radius r_i. At very small radius r_1, the homology of X_1 is same as P, in particular, no holes exist. At radius r_2, three holes appear; one of them is much larger than the others. At radius r_3, the two small holes disappear, but the large one still exists. Finally at radius r_4, the larger hole disappears as well.

defined as

$$X_i := \bigcup_{p \in P} B(p; r_i)$$

for all i, where $B(p; r_i)$ is a disk of center p and radius r_i.

The 1-dimensional homology groups $H_1(X_i)$ can extract holes in the point cloud.

1. Three holes appear in total.
2. Two small holes are born at r_2 and die at r_3.
3. One large hole is born at r_2, still survives at r_3, and dies at r_4.

The robustness of the holes can be expressed by "lifetimes" $r_3 - r_2$ and $r_4 - r_2$ respectively, and since the latter is much greater than the former, we can state that the latter (larger) hole is more robust than the smaller holes which may just be a product of noise.

In order to formalize the above observation, we consider the sequence of maps between homology groups

$$H_1(X_1) \to H_1(X_2) \to H_1(X_3) \to H_1(X_4),$$

where the maps are induced by inclusions. Such a (one-way) sequence of maps between homology groups

$$HX_1 \to HX_2 \to \cdots \to HX_n, \tag{1.1}$$

where $H := H(-; K)$ is a homology functor with coefficient field K, is called a *persistent homology group*.

To deal with a persistent homology group and extract its generators, we need some algebraic framework. The earliest such work was the paper [28], in which the authors formalized a persistent homology group as a module over a graded ring and decomposed

it into indecomposable submodules. By the decomposition theorem [28, Corollary 3.1], it follows that each indecomposable submodule is uniquely expressed by a pair of birth and death values.

The subsequent important work is the paper [5], which introduced representation theory of quivers for the formalization of persistent homology, and this is the main framework of this thesis. A quiver is a directed graph, and its representation is a diagram of the shape of the quiver, composed of vector spaces and linear maps. In particular, the persistent homology group (1.1) is a representation of the one-way quiver

$$\overset{1}{\circ} \longrightarrow \overset{2}{\circ} \longrightarrow \cdots \longrightarrow \overset{n}{\circ}.$$

This abstraction can naturally generalize persistent homology for a large selection of underlying quivers, as explained in the following. Such generalized persistent homology groups are called *persistence modules*.

This thesis presents results for two topics. The following sections state the results of earlier research, as well as our contributions to each topic. Our guiding question throughout is "How can we capture any topological invariant in a deformation of a point cloud?". The author faced this question in topological analysis of granular materials [26] as demonstrated in Appendix A.

1.1. Persistence Modules on Commutative Ladders

Let us consider the two similar point clouds P and P' in Figure 1.3. One can regard

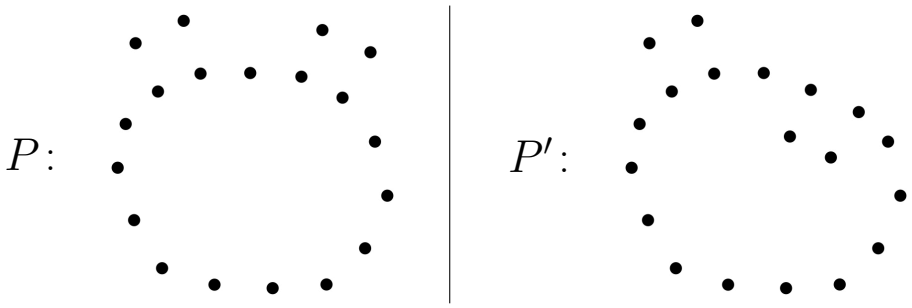


Figure 1.3.: Two point clouds P and P'. These are almost the same, but the difference exists in the upper right part.

these as a deformation from P to P'.

Zigzag persistence modules [5] enable us to look into topological similarity in these point clouds. Fixing a radius r_2 as defined in Figure 1.2, topological spaces $X_2 := \bigcup_{p\in P} B(p; r_2)$ and $Y_2 := \bigcup_{p\in P'} B(p; r_2)$ are obtained (see Figure 1.4).

Although these topological spaces naturally form no filtration, their union can be used to form the diagram

$$X_2 \xrightarrow{\subset} X_2 \cup Y_2 \xleftarrow{\supset} Y_2.$$

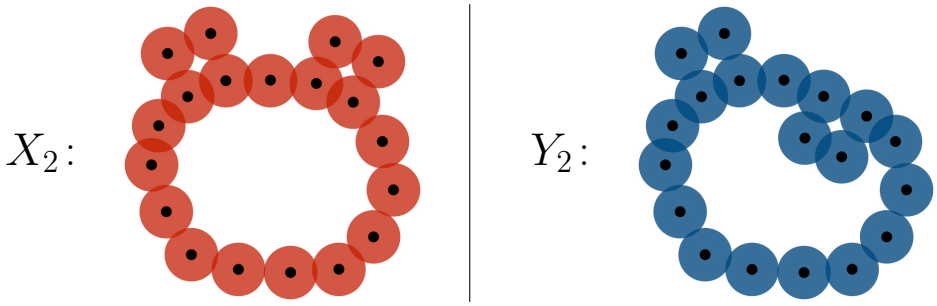


Figure 1.4.: Topological spaces X_2 and Y_2 are unions of disks. These are generated from the point clouds P and P' and the fixed radius r_2.

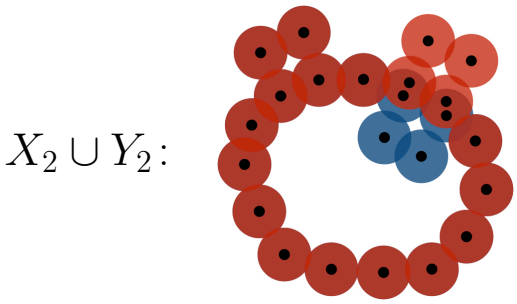


Figure 1.5.: The union space $X_2 \cup Y_2$. The space X_2 is superimposed on Y_2. One large hole and three tiny holes are observed.

The homology functor induces the representation

$$H_1(X_2) \longrightarrow H_1(X_2 \cup Y_2) \longleftarrow H_1(Y_2),$$

which is an example of a zigzag persistence module. In the language of representation theory of quivers, this representation is decomposed into birth-death pairs, that is, it is isomorphic to the direct sum of representations $\bigoplus_{i=1}^{4} I_i$ where the indecomposables I_i are of the forms

$$\begin{aligned} I_1 &: K \xrightarrow{\mathrm{id}_K} K \xleftarrow{\mathrm{id}_K} K \\ I_2 &: K \xrightarrow{\mathrm{id}_K} K \xleftarrow{\mathrm{id}_K} K \\ I_3 &: K \xrightarrow{\mathrm{id}_K} K \longleftarrow 0, \text{ and} \\ I_4 &: 0 \longrightarrow K \xleftarrow{\mathrm{id}_K} K. \end{aligned} \tag{1.2}$$

Such a representation I_i is called an *interval representation*. Table 1.1 shows the correspondence between the bases of the vector spaces and the holes in the spaces. In essence, shared topological structures are given by the holes corresponding to I_1 and I_2, not I_3 nor I_4.

However, as presented before this section, the robustness of the holes are different. In particular, I_1 and I_2 are equivalent in spite of their different robustness. To distinguish

	X_2	$X_2 \cup Y_2$	Y_2
I_1	the large hole	the large hole	the large hole
I_2	the upper left tiny hole	the upper left tiny hole	the upper left tiny hole
I_3	the upper right tiny hole	the upper right tiny hole	
I_4		the inside tiny hole	the inside tiny hole

Table 1.1.: Correspondences between interval representations and holes in the topological spaces.

them, we introduce another radius r_3 in Figure 1.2, and construct topological spaces $X_3 := \bigcup_{p \in P} B(p; r_3)$, $Y_3 := \bigcup_{p \in P'} B(p; r_3)$, as well as their union (see Figure 1.6).

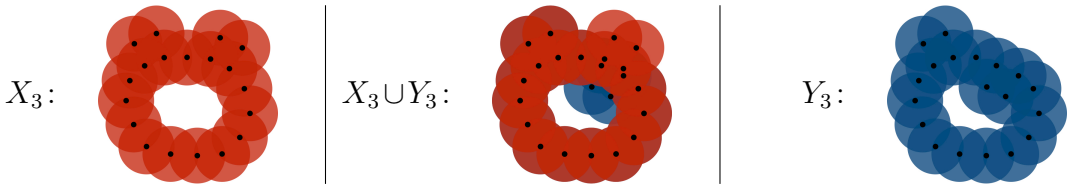


Figure 1.6.: Three topological spaces X_3, Y_3, and their union. X_3 and Y_3 are generated from the point clouds P and P' and the fixed radius r_3.

In this case, the induced representation is

$$H_1(X_3) \longrightarrow H_1(X_3 \cup Y_3) \longleftarrow H_1(Y_3),$$

which is isomorphic to I_1. To study the representation at r_3 and that at r_2 simultaneously, let us consider the following representation of a more complicated quiver.

$$\begin{array}{ccccc} H_1(X_3) & \longrightarrow & H_1(X_3 \cup Y_3) & \longleftarrow & H_1(Y_3) \\ \uparrow & & \uparrow & & \uparrow \\ H_1(X_2) & \longrightarrow & H_1(X_2 \cup Y_2) & \longleftarrow & H_1(Y_2) \end{array} \tag{1.3}$$

Every morphism is induced by a canonical inclusion. This representation is decomposed into

$$J_1: \begin{array}{ccccc} K & \xrightarrow{\mathrm{id}_K} & K & \xleftarrow{\mathrm{id}_K} & K \\ {\scriptstyle \mathrm{id}_K}\uparrow & & {\scriptstyle \mathrm{id}_K}\uparrow & & {\scriptstyle \mathrm{id}_K}\uparrow \\ K & \xrightarrow{\mathrm{id}_K} & K & \xleftarrow{\mathrm{id}_K} & K \end{array}, \qquad J_2: \begin{array}{ccccc} 0 & \longrightarrow & 0 & \longleftarrow & 0 \\ \uparrow & & \uparrow & & \uparrow \\ K & \xrightarrow{\mathrm{id}_K} & K & \xleftarrow{\mathrm{id}_K} & K \end{array},$$

$$J_3: \begin{array}{ccccc} 0 & \longrightarrow & 0 & \longleftarrow & 0 \\ \uparrow & & \uparrow & & \uparrow \\ K & \xrightarrow{\mathrm{id}_K} & K & \longleftarrow & 0 \end{array}, \text{ and } \qquad J_4: \begin{array}{ccccc} 0 & \longrightarrow & 0 & \longleftarrow & 0 \\ \uparrow & & \uparrow & & \uparrow \\ 0 & \longrightarrow & K & \xleftarrow{\mathrm{id}_K} & K \end{array}.$$

For $i = 1, 2, 3$, and 4, J_i respectively corresponds to I_i in (1.2). Now robustness of the generators is expressed in the vertical direction. Consequently, the robustness allows us to distinguish generators corresponding to J_1 and J_2.

In general, a representation of the shape of the diagram (1.3) is called a *persistence module on the commutative ladder* $CL(fb)$. The direct summands isomorphic to J_1 give robust homology generators common in two point clouds.

The paper [16] introduced such a perspective, presented a general definition of commutative ladders $CL(\tau_n)$, and gave a finiteness theorem (see Definition 2.2.3 and Theorem 2.2.4). Moreover, the paper provided a decomposition algorithm for persistence modules on $CL(fb)$ but nothing else.

Contributions

In Chapter 3, we provide a new framework using arrow categories for decompositions of persistence modules on commutative ladders $CL(\tau_n)$ of finite type $n \leq 4$.

Theorem 3.1.1 redefines every persistence module as a homomorphism between representations of an A_n type quiver. Thanks to this identification and Lemma 3.1.3 concerning dimension properties of the homomorphism spaces, each homomorphism is converted to a matrix, as presented in Definition 3.1.5. As with the decompositions in the category of representations, the matrix is decomposed into indecomposables using isomorphisms in the arrow category, to which the matrix belongs. The isomorphisms in the arrow category can also be written in matrix form. As in linear algebra theory, the transformation by an isomorphism is performed by a combination of elementary row and column operations of the matrix. However, not all of the elementary operations are permissible. We remark on this point in Subsection 3.1.3.

The decomposition algorithm of the matrix is given in Algorithm 1. This algorithm is presented in the form of a solver software. The input problem is a matrix together with permissible operations, and the solved output is its decomposed form. Theorem 3.2.3 states that, if the input matrix and permissible operations are constructed from a persistence module on a commutative ladder of finite type, then the algorithm terminates in finite steps and the submatrices in the output matrix correspond to the indecomposable direct summands of the persistence module. This algorithm is independent of the orientation τ_n of the commutative ladder. In other words, this is a unified decomposition algorithm, hence this point is the main advantage against that of the earlier study [16].

1.2. Topology of Sampled Maps

Again let us consider the point clouds P and P' in Figure 1.3. In this section, we treat these as a deformation from P to P'. Suppose the situation that every point in the point cloud is labeled and tracked during the deformation. In this case, the deformation is formalized as a map $f_P \colon P \to P'$.

While we have studied the topology of point clouds P or P', how about that of f_P? We below introduce the theory of homology induced maps of correspondences [21], which

is relevant to this problem.

Consider the following problem.

Problem 1.2.1. Let X and Y be topological spaces, and $f\colon X \to Y$ be a continuous map. If we know only X, Y, and sampling data $f|_S$, which is a restriction of f on a finite subset $S \subset X$, then can we retrieve any information about the homology induced map $f_*\colon HX \to HY$?

Translating this problem to our language, the sample S is P, and we assume the existence of underlying topological spaces X, Y, and a continuous map f. Under these assumptions, the aim is to extract information about the homology induced map f_* only from the map between point clouds $f_P\colon P \to P'$.

The map $f|_S$ is called a *sampled map* of f. The paper [21] suggests the following topological analysis for the sampled map. A *grid* $\mathcal{X}$ of X is a finite collection of subsets of X with disjoint interiors such that $\bigcup \mathcal{X} := \bigcup_{X' \in \mathcal{X}} X' = X$. First, we divide the topological spaces X and Y into grids $\mathcal{X}$ and $\mathcal{Y}$, and let F be the union of regions which have elements of the sample $\mathrm{Gr}(f|_S) := \{\, (s, f(s)) \mid s \in S \,\}$. That is,

$$F := \{\, (x,y) \in X \times Y \mid x \in \exists X' \in \mathcal{X}, y \in \exists Y' \in \mathcal{Y}, (X' \times Y') \cap \mathrm{Gr}(f|_S) \neq \emptyset \,\},$$

the purple regions in Figure 1.8. Thus, we can construct an approximation of the graph $\mathrm{Gr}(f)$ from the sampled map by this subspace, which is called a correspondence.

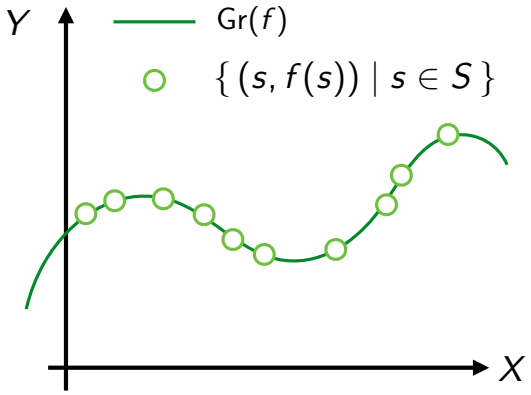


Figure 1.7.: The graph $\mathrm{Gr}(f)$ of f and the graph of the sampled map.

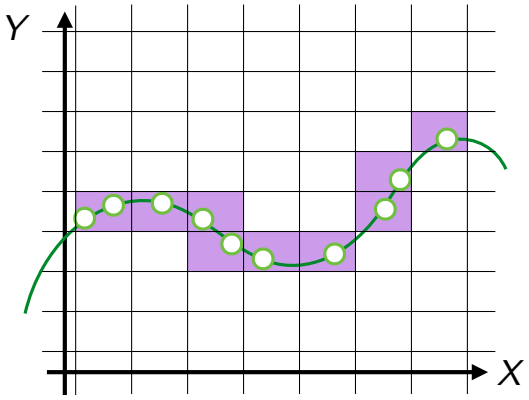


Figure 1.8.: Both spaces are divided into grids and get a correspondence F, which approximates the graph $\mathrm{Gr}(f)$.

Definition 1.2.2. A *correspondence* F from X to Y is a subspace of $X \times Y$.

Definition 1.2.3. For a correspondence F, a homology functor H induces a diagram $HX \xleftarrow{p_*} HF \xrightarrow{q_*} HY$ from a diagram $X \xleftarrow{p} F \xrightarrow{q} Y$, whose p and q are the canonical projections of $X \times Y$. If p_* and q_* satisfy the two properties

- $\operatorname{Im} p_* = HX$ (homologically complete)
- $q_*(\operatorname{Ker} p_*) = 0$ (homologically consistent),

then the *induced map* of F defined as $F_* := q_* \circ p_*^{-1} \colon HX \to HY$ is well-defined.

The graph $\operatorname{Gr}(f)$ of f is a correspondence, hence $\operatorname{Gr}(f)_*$ is defined. We remark that this induced map coincides with f_*. By the following two main theorems in [21], it is guaranteed that F_* restores f_* when the grid is fine and the sample is sufficiently dense.

Theorem 1.2.4 ([21, Theorem 3.10])**.** If a correspondence F satisfies $\operatorname{Gr}(f) \subset F$ and is homologically consistent, then F_* is well-defined and $f_* = F_*$.

Theorem 1.2.5 ([21, Theorem 4.6])**.** If a correspondence F is homologically consistent, a correspondence G is homologically complete, and $G \subset F$, then F_* and G_* are well-defined and $G_* = F_*$.

Contributions

In Section 4.1, we will redefine induced maps of correspondences within the framework of quiver representations. The indecomposable decompositions of quiver representations

$$(HX \xleftarrow{p_*} HF \xrightarrow{q_*} HY) \cong \bigoplus_{1 \le b \le d \le 3} \mathbb{I}[b,d]^{m_{b,d}},$$

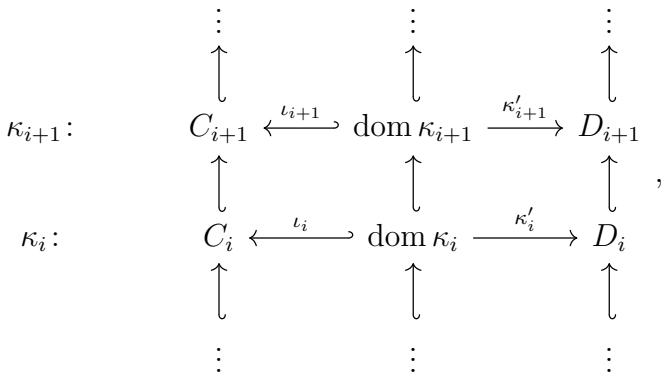
and restriction to the most persistent intervals $\mathbb{I}[1,3]$, give us an assignment among the bases of HX, HF, and HY, which defines the induced map from HX to HY. As presented in Theorem 4.1.1, the definition is an extension of the previous definition, and the setup provides more concise proofs of Theorems 1.2.4 and 1.2.5.

In Section 4.2, we provide persistent homology of sampled maps. In this study, we consider a slightly different problem.

Problem 1.2.6. Let $f \colon X \to Y$ be a continuous map for $X, Y \subset \mathbb{R}^n$. If X, Y, and f are unknown, and we know only a sampled map $f|_S \colon S \to f(S)$ which is a restriction of f on a finite subset $S \subset X$, then can we retrieve any information about the homology induced map $f_* \colon HX \to HY$?

This problem was studied by the paper [14], which presents a method constructing a filtration of simplicial maps from the sampled map. Here we construct a similar filtration

of the sampled map $f\restriction_S$,



where each pair of simplicial maps $C_i \xhookleftarrow{\iota_i} \operatorname{dom}\kappa_i \xrightarrow{\kappa'_i} D_i$ is seen as a filter of the filtration, which captures topological persistence of the original map f. Using the homology functor, this filtration of the sampled map induces the representation presented as the diagram (4.2). Decomposition and restriction to $\mathbb{I}[1,3]$ for all filters define a persistent homology of the sampled map. The well-definedness is not trivial but guaranteed by Theorem 4.2.3.

The main theorem of this study is a stability theorem for this process, Theorem 4.3.4 in Section 4.3, which states that this mapping from input (sampled maps) to output (persistent homology groups) is a 1-Lipschitz map. Moreover, in Subsection 4.4.1, we apply this persistence analysis to the above gridded setting, which has a similar stability theorem (Theorem 4.4.1) as well.

Finally, we approach 2-D persistence modules in Subsection 4.4.2 using the above ideas, and show some numerical results in Section 4.5.

Funding Acknowledgements

These works were supported by JSPS KAKENHI Grant Number 16J03138 and JST CREST Mathematics 15656429.

Chapter 2.

Preliminaries

This chapter collects mathematical notions which we will use later. Throughout this thesis, scalars of vector spaces and coefficient rings of homology groups are a fixed field K. Hence, every homology group is a vector space. Moreover, we assume that every vector space is finite dimensional.

2.1. Notions in Representation Theory of Quivers

2.1.1. Quivers and Representations

A *quiver* $Q = (Q_0, Q_1, s, t)$ (or simply (Q_0, Q_1)) is a directed graph with a set of vertices Q_0, a set of arrows Q_1, and morphisms $s, t\colon Q_1 \to Q_0$ identifying the source and the target vertex of an arrow. An arrow $\alpha \in Q_1$ is denoted by $\alpha\colon s(\alpha) \to t(\alpha)$. In this thesis, we assume that Q_0 and Q_1 are finite sets.

A *representation of a quiver* Q, denoted by $M = (M_a, \varphi_\alpha)_{a \in Q_0, \alpha \in Q_1}$ (or simply (M_a, φ_α) or (M, φ)), is a collection of a vector space M_a for each vertex $a \in Q_0$ and a linear map $\varphi_\alpha\colon M_a \to M_b$ for each arrow $\alpha\colon a \to b \in Q_1$.

Definition 2.1.1. The representation category $\mathrm{rep}(Q)$ of a quiver Q is the following category.

Objects All representations of Q.

Morphisms A morphism from a representation $M = (M_a, \varphi_\alpha)$ to a representation $N = (N_a, \psi_\alpha)$ is defined by

$$f := \{\, f_a\colon M_a \to N_a \mid a \in Q_0 \,\} : M \to N$$

with commutativity

$$\forall \alpha\colon a \to b \in Q_1, \quad \begin{array}{ccc} N_a & \xrightarrow{\psi_\alpha} & N_b \\ {\scriptstyle f_a}\uparrow & & \uparrow{\scriptstyle f_b} \\ M_a & \xrightarrow{\varphi_\alpha} & M_b \end{array} . \tag{2.1}$$

Composition The composition of morphisms $f = \{\, f_a \,\} : M \to N$ and $g = \{\, g_a \,\} : N \to L$ is $gf = \{\, g_a f_a \,\} : M \to L$.

The *zero representation* is the representation with the zero vector space for each vertex and the zero morphism for each arrow. The *direct sum* $M \oplus N$ of representations $M = (M_a, \varphi_\alpha)$ and $N = (N_a, \psi_\alpha)$ of Q is the representation with the vector space $M_a \oplus N_a$ for each vertex $a \in Q_0$ and the linear map $\varphi_\alpha \oplus \psi_\alpha \colon M_a \oplus N_a \to M_b \oplus N_b$ for each arrow $\alpha \colon a \to b \in Q_1$.

A nonzero representation M is said to be *indecomposable*, if M_1 or M_2 is the zero representation for any decomposition $M \cong M_1 \oplus M_2$. It is known that every representation can be uniquely decomposed into a direct sum of indecomposables, unique up to isomorphism and permutations. A quiver is of *finite type* if the number of distinct isomorphism classes of indecomposables is finite, and is of *infinite type* otherwise.

We use the *Auslander–Reiten quiver* $\Gamma = (\Gamma_0, \Gamma_1)$ (or abbreviating *AR quiver*) to list the indecomposables of a quiver, defined as follows. Γ_0 is a set of isomorphism classes of indecomposables, and an arrow $[M] \to [N] \in \Gamma_1$ exists if and only if an irreducible morphism [1] $f \colon M \to N$ exists.

Let $\tau_n = \tau_n^1 \tau_n^2 \cdots \tau_n^{n-1}$ be a sequence of $n-1$ symbols f and b, namely, either $\tau_n^i = f$ or $\tau_n^i = b$. An $A_n(\tau_n)$ *type quiver* (or simply an A_n *type quiver*) is a quiver with the following shape:

$$A_n(\tau_n) : \ \overset{1}{\circ} \longleftrightarrow \overset{2}{\circ} \longleftrightarrow \cdots \longleftrightarrow \overset{n}{\circ}$$

where the i-th $\longleftrightarrow$ is a **f**orward arrow $\longrightarrow$ if $\tau_n^i = f$, or a **b**ackward arrow $\longleftarrow$ if $\tau_n^i = b$. τ_n is called the *orientation* of the A_n type quiver. Gabriel's theorem [18] shows that every indecomposable A_n type representation is isomorphic to an *interval (representation)*

$$\mathbb{I}[b,d] : 0 \longleftrightarrow \cdots \longleftrightarrow 0 \longleftrightarrow \overset{b\text{-th}}{K} \overset{\mathrm{id}_K}{\longleftrightarrow} K \overset{\mathrm{id}_K}{\longleftrightarrow} \cdots \overset{\mathrm{id}_K}{\longleftrightarrow} \overset{d\text{-th}}{K} \longleftrightarrow 0 \longleftrightarrow \cdots \longleftrightarrow 0.$$

Hence, every A_n type representation M can be uniquely decomposed into a direct sum of intervals

$$M \cong \bigoplus_{1 \le b \le d \le n} \mathbb{I}[b,d]^{m_{b,d}} \quad (m_{b,d} \in \mathbb{Z}_{\ge 0} \colon \text{multiplicity}). \tag{2.2}$$

We remark that the same symbols $\mathbb{I}[b,d]$ are used for the intervals of every $A_n(\tau_n)$.

2.1.2. Bound Quivers

The above definition of representations of quivers is sufficient to deal with persistent homology defined later, but in some cases like our work, it can be useful to restrict the class of representations. The concept of bound quivers described below provides such a useful framework for us.

A *path* $(a|\alpha_1\alpha_2\cdots\alpha_\ell|b)$ *from a source* $a \in Q_0$ *to a target* $b \in Q_0$ is a series of arrows $\alpha_1, \ldots, \alpha_\ell$ satisfying $s(\alpha_1) = a$, $t(\alpha_\ell) = b$, and $t(\alpha_i) = s(\alpha_{i+1})$ for all $i \in \{1, \ldots, \ell - 1\}$. The number ℓ is called its *length*. Paths of length 0 are allowed and are denoted by

[1] A morphism $f \colon M \to N$ is irreducible if and only if f is neither a section nor a retraction, and every decomposition $f = f_1 f_2$ implies f_1 is a retraction or f_2 is a section.

$e_a = (a||a)$. The *path algebra* KQ of a quiver Q is a K-vector space freely generated by all paths, with multiplication structure given as

$$(a|\alpha_1 \cdots \alpha_\ell|b)(c|\beta_1 \cdots \beta_m|d) = \begin{cases} (a|\alpha_1 \cdots \alpha_\ell\beta_1 \cdots \beta_m|d) & (b = c), \\ 0 & (\text{otherwise}). \end{cases}$$

An element $\rho = \Sigma_{i=1}^m \lambda_i w_i \in KQ$ where $\lambda_i \in K$ and all w_i have the same source and target is called a *relation* of Q. A *bound quiver* (Q, P) is a pair consisting of a quiver Q and a set of relations $P = \{ \rho_1, \ldots, \rho_m \}$.

We define the *evaluation* of $M = (M_a, \varphi_\alpha)$ on a path $w = (a|\alpha_1 \cdots \alpha_\ell|b)$ to be $\varphi_w = \varphi_{\alpha_\ell} \cdots \varphi_{\alpha_1} \colon M_a \to M_b$. A representation M is a *representation of a bound quiver* (Q, P) if $\varphi_\rho := \Sigma_{i=1}^m \lambda_i \varphi_{w_i} = 0$ for all relations $\rho = \Sigma_{i=1}^m \lambda_i w_i \in P$.

Our work requires the following commutativity relations. A relation of the form $w - w'$ is called a *commutativity relation*, where w and w' are two different paths sharing the same source and target. To understand it intuitively, one can consider the following bound quiver (Q, P):

$$Q = \begin{array}{ccc} a & \xrightarrow{\alpha} & b \\ {\scriptstyle\beta}\uparrow & & \uparrow{\scriptstyle\delta} \\ c & \xrightarrow{\gamma} & d \end{array}, \quad P = \{ \rho = \beta\alpha - \gamma\delta \}.$$

By definition, every representation (M, φ) of the bound quiver (Q, P) satisfies $\varphi_\rho = \varphi_\alpha\varphi_\beta - \varphi_\delta\varphi_\gamma = 0$. Therefore, the diagram of the representation

$$\begin{array}{ccc} M_a & \xrightarrow{\varphi_\alpha} & M_b \\ {\scriptstyle\varphi_\beta}\uparrow & & \uparrow{\scriptstyle\varphi_\delta} \\ M_c & \xrightarrow{\varphi_\gamma} & M_d \end{array}$$

must be commutative.

In the same way as of representations of quivers, various objects of representations of bound quivers are also defined; morphisms, isomorphisms, direct sums, a zero representation, indecomposability, finite (or infinite) type, and AR quivers. Under such definitions, the representation category $\mathrm{rep}(Q, P)$ of a bound quiver (Q, P) is composed in the natural way. This is the full subcategory of $\mathrm{rep}(Q)$ whose objects are the representations of (Q, P).

2.2. Notions in Topological Data Analysis

2.2.1. Simplicial Complexes for Point Clouds

In some studies such as in this thesis, persistent homology is defined for a filtration. Before presenting the definition, let us give two typical constructions of a filtration from a point cloud in preparation for the theory in Chapter 4.

Let P be a point cloud in $\mathbb{R}^n$. An *abstract simplicial complex* Ω for P is a collection of subsets of P such that

$$P_1 \subset P_2 \text{ and } P_2 \in \Omega \Longrightarrow P_1 \in \Omega.$$

Definition 2.2.1. The *Čech complex* C_r for P with a radius r is an abstract simplicial complex defined as

$$C_r = \{\, \sigma \subset P \mid \bigcap_{p \in \sigma} B(p; r) \neq \emptyset \,\},$$

where $B(p; r)$ is the ball of center p and radius r. Let $d_{\mathbb{R}^n}$ be the Euclidean metric on $\mathbb{R}^n$. The *Vietoris–Rips complex* V_r for P with a radius r is an abstract simplicial complex defined as

$$V_r = \{\, \sigma \subset P \mid \forall p_1, p_2 \in \sigma, d_{\mathbb{R}^n}(p_1, p_2) \leq r \,\}$$ [2].

These complexes naturally form filtrations by increasing the radius parameter r. For example, increasing radius parameters $r_1 < r_2 < \cdots < r_n$ induce the filtration of Čech complexes

$$C_{r_1} \subset C_{r_2} \subset \cdots \subset C_{r_n}.$$

Filtrations of Čech complexes and Vietoris–Rips complexes are used for capturing topological shapes of point clouds.

Čech complexes have a theoretically good property, known as *Nerve Theorem* [4]. On the other hand, Vietoris–Rips complexes are computationally lighter, hence we select them depending on the situation (see [13, Section III.2] for details). Such filtrations using abstract simplicial complexes are frequently used for analysis of data given in the form of point clouds.

2.2.2. Persistent Homology, Persistence Diagrams, and Persistence Modules

Definition 2.2.2. Consider a filtration of spaces (e.g. topological spaces, simplicial complexes)

$$X : X_1 \subset X_2 \subset \cdots \subset X_n.$$

The *persistent homology group* of the filtration X is the sequence

$$HX : HX_1 \to HX_2 \to \cdots \to HX_n \tag{2.3}$$

induced by the homology functor H.

[2] Ordinarily, Vietoris–Rips complexes are defined not by the inequality $d_{\mathbb{R}^n}(p_1, p_2) \leq r$ but by $d_{\mathbb{R}^n}(p_1, p_2) \leq 2r$. We adopt this definition for the comparison of numerical results in Section 4.5 with an earlier research [14].

In the context of quiver representations, the persistent homology is regarded as a representation of an $A_n(ff\cdots f)$ type quiver. Each direct summand $\mathbb{I}[b,d]$ in the unique decomposition

$$HX \cong \bigoplus_{1\le b\le d\le n} \mathbb{I}[b,d]^{m_{b,d}} \quad (m_{b,d}\in\mathbb{Z}_{\ge 0}\text{: multiplicity})$$

corresponds to a generator of a homology group which is born at HX_b and dies at HX_{d+1}. Therefore, the length $d-b$ of an interval is called its *lifetime* or *persistence*.

The *persistence diagram* is a multiset

$$\{\,(b,d)\mid 1\le b\le d\le n, (b,d) \text{ has multiplicity } m_{b,d}\,\},$$

determined by the decomposition. We illustrate it by plotting it on a plane[3]. This description gives us an overview of the generators of all homology groups and therefore, this approach is frequently used for applications of persistent homology.

Furthermore, the framework of quiver representations can extend persistent homology. We call representations of (bound) quivers *persistence modules*. Zigzag persistence modules [5] are an example of the extension, enabling to capture persistent topological features in deformations of spaces.

Consider deformations of topological spaces $(X_1,\dots,X_T)$, a sequence of topological spaces. Although the sequence is not a filtration in general, the unions of neighboring spaces can generate the sequence of their canonical inclusions

$$X_1 \hookrightarrow X_1\cup X_2 \hookleftarrow X_2 \hookrightarrow \cdots \hookrightarrow X_{T-1}\cup X_T \hookleftarrow X_T.$$

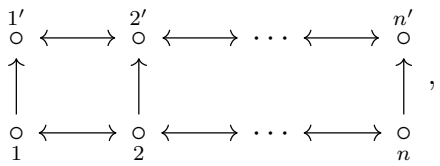
The *zigzag persistence module* of $(X_1,\dots,X_T)$ is given by the sequence of homology

$$H(X_1)\to H(X_1\cup X_2)\leftarrow H(X_2)\to\cdots\to H(X_{T-1}\cup X_T)\leftarrow HX_T.$$

The decomposition of the zigzag persistence module as a $A_{2T-1}(fbfb\cdots fb)$ type representation yields a persistence diagram again, where each interval captures the persistence of a homology generator in the deformations of spaces.

2.2.3. Persistence Modules on Commutative Ladders

Definition 2.2.3. Let $L(\tau_n)$ denote a quiver of the form



where the upper and lower quivers are the same $A_n(\tau_n)$ type. Let C be the set of all possible commutativity relations. A *commutative ladder* $CL_n(\tau_n)$ (or simply $CL(\tau_n)$) is a bound quiver $(L(\tau_n), C)$. *Persistence modules on commutative ladders* are representations of the bound quiver $CL(\tau_n)$.

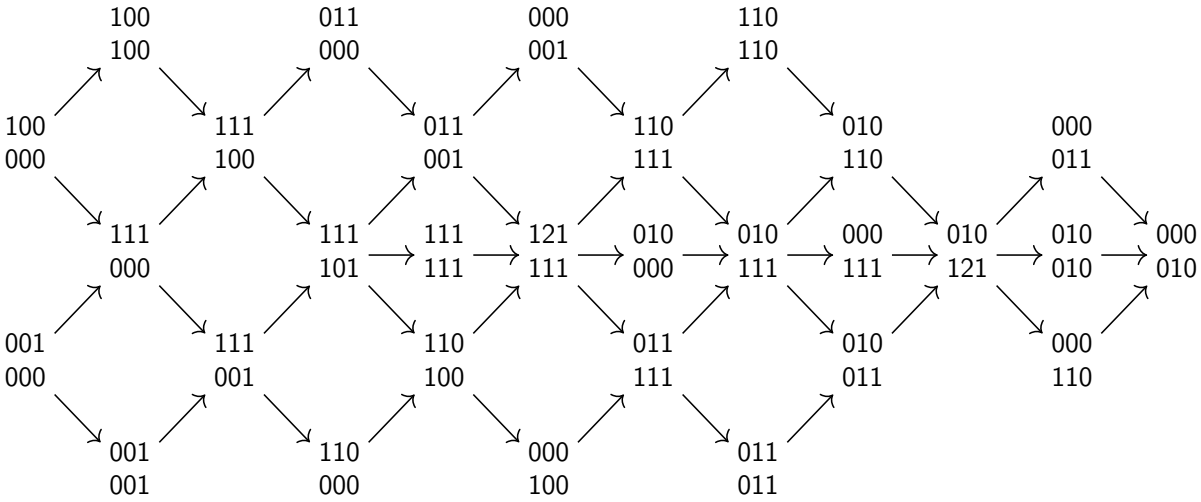


Figure 2.1.: Auslander–Reiten quiver of the commutative ladder $CL_3(bf)$ [16].

The original paper [16] of persistence modules on $CL(\tau_n)$ described the following finiteness theorem.

Theorem 2.2.4 ([16, Theorem 3 and 4])**.** For any orientation τ_n,

$$CL(\tau_n) \text{ is of finite type } \iff n \leq 4.$$

Example 2.2.5. As an example of commutative ladders of finite type, one can present the AR quiver of the commutative ladder $CL_3(bf)$ in Figure 2.1. The vertices of the AR quiver are expressed in the form of dimension vectors. The *dimension vector* of a representation M is a vector of dimensions of $M(a)$ over K for vertices $a \in Q_0$. For convenience, the vectors are aligned according to the positions of the vertices $a \in Q_0$. For example, the dimension vector of a representation

$$\begin{array}{ccccc} K & \longleftarrow & K^2 & \longrightarrow & K^3 \\ \uparrow & & \uparrow & & \uparrow \\ K^3 & \longleftarrow & K^2 & \longrightarrow & K \end{array}$$

is $\begin{smallmatrix}123\\321\end{smallmatrix}$. We remark that for the isomorphism class of a representation, the dimension vector is uniquely determined (but not vice versa). In analogy with the interval representations of A_n quivers, representatives of the indecomposables in Figure 2.1 expressed by the dimension vectors not including 2 can be chosen as follows. The morphisms in the indecomposables are identity maps from K to K, or 0 otherwise. Exceptions are $\begin{smallmatrix}121\\111\end{smallmatrix}$ and

[3]This image is also called the persistence diagram, particularly in real data analysis.

$\begin{smallmatrix}010\\121\end{smallmatrix}$. These are isomorphic classes of

$$\begin{array}{ccccc} K & \xleftarrow{[1\ 0]} & K^2 & \xrightarrow{[0\ 1]} & K \\ {\scriptstyle 1}\uparrow & & \left[\begin{smallmatrix}1\\1\end{smallmatrix}\right]\uparrow & & \uparrow{\scriptstyle 1} \\ K & \xleftarrow{1} & K & \xrightarrow{1} & K \end{array} \quad \text{and} \quad \begin{array}{ccccc} 0 & \longleftarrow & K & \longrightarrow & 0 \\ \uparrow & & {\scriptstyle [1\ 1]}\uparrow & & \uparrow \\ K & \xleftarrow{[1\ 0]} & K^2 & \xrightarrow{[0\ 1]} & K \end{array} \tag{2.4}$$

respectively. For convenience, we fix these representatives throughout this thesis.

The notion of persistence diagrams has been extended to $CL(\tau_n)$ by using indecomposable decomposition. Let $\Gamma = (\Gamma_0, \Gamma_1)$ be the AR quiver of $CL(\tau_n)$, then a representation M of $CL(\tau_n)$ has the decomposition

$$M \cong \bigoplus_{[I]\in\Gamma_0} I^{m_{[I]}} \quad (m_{[I]} \in \mathbb{Z}_{\geq 0}\text{: multiplicity}).$$

The *persistence diagram* of M is the map $\Gamma_0 \ni [I] \mapsto m_{[I]} \in \mathbb{Z}_{\geq 0}$. If $CL(\tau_n)$ is of finite type, then Γ_0 is a finite set, hence we can draw the persistence diagram by putting the numbers $m_{[I]}$ on the corresponding vertices in Γ. The paper [16] suggested the utility of such persistence diagrams for application to real data analysis.

2.3. Arrow Categories

Finally, we provide the definition of arrow categories. This notion will contribute to a decomposition algorithm in Chapter 3.

Definition 2.3.1. Let $\mathcal{C}$ be a category. The *arrow category* $\mathrm{arr}(\mathcal{C})$ of $\mathcal{C}$ is the following category.

Objects All morphisms $f \in \mathrm{Hom}(C_1, C_2)$ for all objects C_1 and C_2 of $\mathcal{C}$.

Morphisms A morphism $\varphi\colon f \to g$ from $f \in \mathrm{Hom}(C_1, C_2)$ to $g \in \mathrm{Hom}(D_1, D_2)$ is a pair of morphisms $(\varphi_1\colon C_1 \to D_1, \varphi_2\colon C_2 \to D_2)$ satisfying the commutativity

$$\begin{array}{ccc} D_1 & \xrightarrow{g} & D_2 \\ {\scriptstyle \varphi_1}\uparrow & & \uparrow{\scriptstyle \varphi_2} \\ C_1 & \xrightarrow{f} & C_2 \end{array}.$$

Composition The composition of morphisms $\varphi = (\varphi_1, \varphi_2)\colon f \to g$ and $\psi = (\psi_1, \psi_2)\colon g \to h$ is $\psi\varphi := (\psi_1\varphi_1, \psi_2\varphi_2)$.

Chapter 3.

Matrix Method for Persistence Modules on Commutative Ladders of Finite Type

This chapter serves a decomposition algorithm for persistence modules on commutative ladders of finite type. Since the algorithm consists only of elementary operations on matrices, one can implement it smoothly and effectively. The content of this chapter is based on the published paper:

[1] H. Asashiba, E. G. Escolar, Y. Hiraoka, and H. Takeuchi, Matrix Method for Persistence Modules on Commutative Ladders of Finite Type, Jpn. J. Ind. Appl. Math. DOI:10.1007/s13160-018-0331-y

The final publication is available at https://link.springer.com/article/10.1007/s13160-018-0331-y. This paper provides proofs only for the orientations f, fb, and fff, because those for the other orientations are similar. Therefore in this thesis, we present proofs for the other orientations b, bf, and bfb, to support the above paper.

3.1. Matrix Notation

3.1.1. From Representations to Arrows

Theorem 3.1.1. Let τ_n be an orientation of an $A_n(\tau_n)$ quiver. There is an isomorphism of categories

$$\operatorname{rep} CL(\tau_n) \cong \operatorname{arr}(\operatorname{rep} A_n(\tau_n)).$$

Proof. An isomorphism functor $F\colon \operatorname{rep} CL(\tau_n) \to \operatorname{arr}(\operatorname{rep} A_n(\tau_n))$ can be constructed by taking a persistence module $M \in \operatorname{rep} CL(\tau_n)$ to the morphism defined by M from its bottom row to its top row. Similarly, a morphism between two persistence modules $\lambda : M \to N$ defines a morphism $F(\lambda)$ between the corresponding arrows $F(M)$, $F(N)$ in the obvious way. □

The isomorphism $F\colon \operatorname{rep} CL(\tau_n) \to \operatorname{arr}(\operatorname{rep} A_n(\tau_n))$ constructed above enables us to identify a persistence module M on $CL(\tau_n)$ with the corresponding arrow $F(M)$.

3.1.2. From Arrows to Matrix Form

Fix an orientation τ_n. We define the following symbol for convenience.

Definition 3.1.2. The relation $\trianglerighteq$ on the set $\{\, \mathbb{I}[b,d] \mid 1 \le b \le d \le n \,\}$ of all interval representations of $A_n(\tau_n)$ is defined as follows.

$$\mathbb{I}[a,b] \trianglerighteq \mathbb{I}[c,d] \iff \mathrm{Hom}(\mathbb{I}[a,b], \mathbb{I}[c,d]) \neq 0.$$

It can be confirmed that $\trianglerighteq$ is reflexive and antisymmetric[1], but in general is not transitive. See Example 3.1.4 below. Since we use the same symbols $\mathbb{I}[b,d]$ for all $A_n(\tau_n)$, the relation $\trianglerighteq$ depends on the underlying orientation τ_n. For instance, one can easily check that $\mathbb{I}[2,2] \trianglerighteq \mathbb{I}[1,2]$ for $\tau_2 = f$, but $\mathbb{I}[2,2] \not\trianglerighteq \mathbb{I}[1,2]$ for $\tau_2 = b$. Here, we use the following expression $\mathbb{I}[a,b] \triangleright \mathbb{I}[c,d]$ if $\mathbb{I}[a,b] \trianglerighteq \mathbb{I}[c,d]$ and $\mathbb{I}[a,b] \neq \mathbb{I}[c,d]$.

Lemma 3.1.3. Let $\mathbb{I}[a,b], \mathbb{I}[c,d]$ be interval representations of $A_n(\tau_n)$.

1. The dimension of $\mathrm{Hom}(\mathbb{I}[a,b], \mathbb{I}[c,d])$ as a K-vector space is either 0 or 1.
2. A K-vector space basis $\left\{ f_{a:b}^{c:d} \right\}$ can be chosen for each nonzero $\mathrm{Hom}(\mathbb{I}[a,b], \mathbb{I}[c,d])$ such that if $\mathbb{I}[a,b] \trianglerighteq \mathbb{I}[c,d]$, $\mathbb{I}[c,d] \trianglerighteq \mathbb{I}[e,f]$ *and* $\mathbb{I}[a,b] \trianglerighteq \mathbb{I}[e,f]$, then
$$f_{a:b}^{e:f} = f_{c:d}^{e:f} f_{a:b}^{c:d}. \tag{3.1}$$

Proof.

1. The notation $[a,b] = \{a, a+1, \ldots, b\}$ is used to denote the interval of integers i with $a \le i \le b$. Consider $g = \{g_i\}_{i=1}^n \in \mathrm{Hom}(\mathbb{I}[a,b], \mathbb{I}[c,d])$ and suppose that $\mathrm{Hom}(\mathbb{I}[a,b], \mathbb{I}[c,d])$ is nonzero.

   Note that $g_i = 0$ for $i \notin [a,b] \cap [c,d]$. It follows that if $[a,b] \cap [c,d] = \emptyset$, then $\mathrm{Hom}(\mathbb{I}[a,b], \mathbb{I}[c,d]) = \{0\}$, a contradiction. Therefore $[a,b] \cap [c,d] \neq \emptyset$.

   Fix an index $j \in [a,b] \cap [c,d] \neq \emptyset$. We claim that $g_i = g_j$ for any $i \in [a,b] \cap [c,d]$, by the commutativity requirement on morphisms. To illustrate this point, suppose that $i = j+1$ with $i \in [a,b] \cap [c,d]$. Then $g_i = g_j$ follows from the commutativity of
$$\begin{array}{ccc} K & \xrightarrow{\mathrm{id}_K} & K \\ {\scriptstyle g_j}\uparrow & & \uparrow{\scriptstyle g_i} \\ K & \xrightarrow{\mathrm{id}_K} & K \end{array} \quad \text{or} \quad \begin{array}{ccc} K & \xleftarrow{\mathrm{id}_K} & K \\ {\scriptstyle g_j}\uparrow & & \uparrow{\scriptstyle g_i} \\ K & \xleftarrow{\mathrm{id}_K} & K \end{array}$$
   for the i-th orientation f or b, respectively. A similar argument demonstrates that the claim presented above holds for $i = j-1$ with $i \in [a,b] \cap [c,d]$. Repeating this argument, one obtains $g_i = g_j$ as long as $i \in [a,b] \cap [c,d]$.

   Consequently, any morphism g is determined uniquely by its value g_j for some $j \in [a,b] \cap [c,d]$. This provides an isomorphism of K-vector spaces
$$\mathrm{Hom}(\mathbb{I}[a,b], \mathbb{I}[c,d]) \cong \mathrm{Hom}_K(K,K)$$

[1] $\mathbb{I}[a,b] \trianglerighteq \mathbb{I}[c,d]$ and $\mathbb{I}[c,d] \trianglerighteq \mathbb{I}[a,b]$ imply $\mathbb{I}[a,b] = \mathbb{I}[c,d]$.

by taking g to g_j.

Since $\mathrm{Hom}_K(K,K) \cong K$, one can conclude that if $\mathrm{Hom}(\mathbb{I}[a,b],\mathbb{I}[c,d])$ is nonzero, then its dimension is 1.

2. For all pairs of intervals with $\mathbb{I}[a,b] \trianglerighteq \mathbb{I}[c,d]$ define $f_{a:b}^{c:d}$ by

$$\left(f_{a:b}^{c:d}\right)_i = \begin{cases} \mathrm{id}_K & \text{if } i \in [a,b] \cap [c,d], \\ 0 & \text{otherwise.} \end{cases}$$

The discussion presented above indicates that $f_{a:b}^{c:d}$ is in $\mathrm{Hom}(\mathbb{I}[a,b],\mathbb{I}[c,d])$, and that any $g \in \mathrm{Hom}(\mathbb{I}[a,b],\mathbb{I}[c,d])$ can be written as $g = g_j(1) f_{a:b}^{c:d}$ for any $j \in [a,b] \cap [c,d]$. Moreover, by construction, this choice of $f_{a:b}^{c:d}$ satisfies Eq. (3.1).

□

Example 3.1.4. With orientation $\tau_n = ff\cdots f$, the homomorphism spaces are

$$\mathrm{Hom}(\mathbb{I}[a,b],\mathbb{I}[c,d]) = \begin{cases} K f_{a:b}^{c:d}, & c \le a \le d \le b, \\ 0, & \text{otherwise.} \end{cases}$$

The basis functions $f_{a:b}^{c:d}$ are given as

$$\left(f_{a:b}^{c:d}\right)_i = \begin{cases} \mathrm{id}_K, & a \le i \le d, \\ 0, & \text{otherwise.} \end{cases}$$

With $n = 2$, $\mathbb{I}[2,2] \trianglerighteq \mathbb{I}[1,2]$ and $\mathbb{I}[1,2] \trianglerighteq \mathbb{I}[1,1]$, but $\mathbb{I}[2,2] \not\trianglerighteq \mathbb{I}[1,1]$. This result also provides an example illustrating that $\trianglerighteq$ may not be transitive.

For convenience, we define $f_{a:b}^{c:d} = 0$ for the case $\mathbb{I}[a,b] \not\trianglerighteq \mathbb{I}[c,d]$.

Now, letting M be a representation of $CL(\tau_n)$ with $n \le 4$, by the isomorphism $F : \mathrm{rep}\, CL(\tau_n) \cong \mathrm{arr}(\mathrm{rep}\, A_n(\tau_n))$ in Theorem 3.1.1, M is identified with its corresponding arrow $F(M) \colon V \to W$ in $\mathrm{arr}(\mathrm{rep}\, A_n(\tau_n))$. Note that V is in $\mathrm{rep}\, A_n(\tau_n)$ and can therefore be decomposed as

$$\eta_V \colon V \cong \bigoplus_{1\le a\le b\le n} \mathbb{I}[a,b]^{m_{a,b}}, \quad (m_{a,b} \in \mathbb{Z}_{\ge 0}\text{: multiplicity}) \tag{3.2}$$

as in Eq. (2.2). A similar isomorphism η_W can be obtained for W. Through these isomorphisms, define

$$\Phi = \eta_W F(M) \eta_V^{-1} \colon \bigoplus_{1\le a\le b\le n} \mathbb{I}[a,b]^{m_{a,b}} \to \bigoplus_{1\le c\le d\le n} \mathbb{I}[c,d]^{m'_{c,d}}. \tag{3.3}$$

In fact, $(\eta_V, \eta_W) : F(M) \to \Phi$ is an isomorphism in $\mathrm{arr}(\mathrm{rep}\, A_n(\tau_n))$.

Moreover, Φ can be written in block matrix form as

$$\Phi = \left[\Phi_{a:b}^{c:d}\right],$$

where each block matrix entry $\Phi_{a:b}^{c:d}\colon \mathbb{I}[a,b]^{m_{a,b}} \to \mathbb{I}[c,d]^{m'_{c,d}}$ is obtained from Φ by the appropriate inclusion and projection. That is, $\Phi_{a:b}^{c:d}$ is the composition of

$$\mathbb{I}[a,b]^{m_{a,b}} \xrightarrow{\iota} \bigoplus_{1\le a\le b\le n} \mathbb{I}[a,b]^{m_{a,b}} \xrightarrow{\Phi} \bigoplus_{1\le c\le d\le n} \mathbb{I}[c,d]^{m'_{c,d}} \xrightarrow{\pi} \mathbb{I}[c,d]^{m'_{c,d}}. \tag{3.4}$$

In a similar manner, each block $\Phi_{a:b}^{c:d}$ can be expressed further as a matrix of homomorphisms

$$\Phi_{a:b}^{c:d} = [g_{i,j}], \quad (1 \le i \le m_{a,b}, 1 \le j \le m'_{c,d}),$$

where each $g_{i,j} \in \mathrm{Hom}(\mathbb{I}[a,b], \mathbb{I}[c,d])$.

For intervals $\mathbb{I}[a,b] \trianglerighteq \mathbb{I}[c,d]$, part two of Lemma 3.1.3 shows that for each i, j one can write $g_{i,j} = \mu_{i,j} f_{a:b}^{c:d}$ for some $\mu_{i,j} \in K$. Factoring out $f_{a:b}^{c:d}$ from $\Phi_{a:b}^{c:d}$ with $\mathbb{I}[a,b] \trianglerighteq \mathbb{I}[c,d]$, we get

$$\Phi_{a:b}^{c:d} = \begin{cases} M_{a:b}^{c:d} f_{a:b}^{c:d} & \text{if } \mathbb{I}[a,b] \trianglerighteq \mathbb{I}[c,d], \\ 0 & \text{otherwise,} \end{cases}$$

where each $M_{a:b}^{c:d}$ is an $m'_{c,d} \times m_{a,b}$ matrix with entries in K. To summarize, we define the following.

Definition 3.1.5. Let M be a persistence module on $CL(\tau_n)$. The block matrix form $\Phi(M)$ of M is

$$\Phi(M) = \left[\Phi_{a:b}^{c:d}\right] = \left[M_{a:b}^{c:d} f_{a:b}^{c:d}\right]_{\mathbb{I}[a,b]\trianglerighteq\mathbb{I}[c,d]},$$

where each $\Phi_{a:b}^{c:d}$ is defined as presented in Eq. (3.4).

In the matrix formalism, we label the rows and columns of the block matrix corresponding to the summand $\mathbb{I}[a,b]^{m_{a,b}}$ by $a{:}b$. We say that the block $\Phi_{a:b}^{c:d}$ is in row $c{:}d$ and column $a{:}b$.

3.1.3. Permissible Operations

Although we have written Φ in block matrix form, not all of the usual row and column operations on K-matrices correspond to a meaningful change of basis. The fact that there exist some pairs of intervals where $\mathrm{Hom}(\mathbb{I}[a,b], \mathbb{I}[c,d])$ is zero leads to some complications.

If $(R,S) : \Phi' \cong \Phi$ is an isomorphism, then

$$\begin{array}{ccc} \displaystyle\bigoplus_{1\le a\le b\le n} \mathbb{I}[a,b]^{m_{a,b}} & \xrightarrow{\Phi} & \displaystyle\bigoplus_{1\le a\le b\le n} \mathbb{I}[a,b]^{m'_{a,b}} \\ \Big\uparrow{\scriptstyle R} & & \Big\uparrow{\scriptstyle S} \\ \displaystyle\bigoplus_{1\le a\le b\le n} \mathbb{I}[a,b]^{m_{a,b}} & \xrightarrow{\Phi'} & \displaystyle\bigoplus_{1\le a\le b\le n} \mathbb{I}[a,b]^{m'_{a,b}} \end{array}$$

commutes and $\Phi' = S^{-1}\Phi R$. Observing that the domain and codomain of R are direct summations, R can be written in a matrix form $R = \left[R_{a:b}^{c:d} f_{a:b}^{c:d}\right]_{\mathbb{I}[a,b]\trianglerighteq\mathbb{I}[c,d]}$ relative to them,

by an argument similar to that made for Φ. Similarly, S can be written in matrix form. It can be checked that $(R, S) : \Phi' \to \Phi$ is an isomorphism if and only if all the diagonal entries $R^{a:b}_{a:b}$ and $S^{a:b}_{a:b}$ are invertible.

Hereinafter, column operations are discussed, assuming S as the identity. In analogy to usual linear algebra, column operations on Φ correspond to a change of interval summands induced by R. Below, fix a column $a{:}b$ and suppose that $\mathbb{I}[a, b] \trianglerighteq \mathbb{I}[c, d]$.

The block entry at row $c{:}d$ in column $a{:}b$ of $\Phi' = \Phi R$ is

$$\begin{aligned}[\Phi R]^{c:d}_{a:b} &= \sum_{\mathbb{I}[a,b] \trianglerighteq \mathbb{I}[e,f] \trianglerighteq \mathbb{I}[c,d]} (M^{c:d}_{e:f} f^{c:d}_{e:f})(R^{e:f}_{a:b} f^{e:f}_{a:b}) \\ &= \left(\sum_{\mathbb{I}[a,b] \trianglerighteq \mathbb{I}[e,f] \trianglerighteq \mathbb{I}[c,d]} M^{c:d}_{e:f} R^{e:f}_{a:b} \right) f^{c:d}_{a:b}\end{aligned}$$

where ΦR is computed as a usual multiplication of block matrices. In the last step, we used the property that $f^{c:d}_{e:f} f^{e:f}_{a:b} = f^{c:d}_{a:b}$ as guaranteed by Lemma 3.1.3. Note that the resulting coefficient of $f^{c:d}_{a:b}$ above involves only addition and multiplication of K-matrices. Furthermore, since $\mathbb{I}[a, b] \trianglerighteq \mathbb{I}[a, b]$, it is equal to

$$\sum_{\mathbb{I}[a,b] \trianglerighteq \mathbb{I}[e,f] \trianglerighteq \mathbb{I}[c,d]} M^{c:d}_{e:f} R^{e:f}_{a:b} = \left(M^{c:d}_{a:b} R^{a:b}_{a:b} + \sum_{\mathbb{I}[a,b] \triangleright \mathbb{I}[e,f] \trianglerighteq \mathbb{I}[c,d]} M^{c:d}_{e:f} R^{e:f}_{a:b} \right).$$

In this form, we see that, apart from a change of basis $R^{a:b}_{a:b}$ within the column $a{:}b$, we also permit the addition of multiples of columns $e{:}f$ with $\mathbb{I}[a, b] \triangleright \mathbb{I}[e, f]$. A similar analysis can be performed for row operations.

For example, one can consider a persistence module M on $CL_2(f)$ corresponding to

$$\Phi = \begin{array}{c} \\ 2{:}2 \\ 1{:}2 \\ 1{:}1 \end{array} \begin{array}{c} \begin{array}{ccc} 2{:}2 & 1{:}2 & 1{:}1 \end{array} \\ \begin{bmatrix} M^{2:2}_{2:2}\mathrm{id}_{\mathbb{I}[2,2]} & 0 & 0 \\ M^{1:2}_{2:2} f^{1:2}_{2:2} & M^{1:2}_{1:2}\mathrm{id}_{\mathbb{I}[1,2]} & 0 \\ 0 & M^{1:1}_{1:2} f^{1:1}_{1:2} & M^{1:1}_{1:1}\mathrm{id}_{\mathbb{I}[1,1]} \end{bmatrix} \end{array}.$$

Since $1{:}1 \not\trianglerighteq 1{:}2$, $1{:}1 \not\trianglerighteq 2{:}2$, $1{:}2 \not\trianglerighteq 2{:}2$, and $2{:}2 \not\trianglerighteq 1{:}1$, the blocks in the corresponding positions are zero. An automorphism R on $V = \mathbb{I}[2, 2]^{m_{2,2}} \oplus \mathbb{I}[1, 2]^{m_{1,2}} \oplus \mathbb{I}[1, 1]^{m_{1,1}}$ as defined above can be written as

$$R = \begin{array}{c} \\ 2{:}2 \\ 1{:}2 \\ 1{:}1 \end{array} \begin{array}{c} \begin{array}{ccc} 2{:}2 & 1{:}2 & 1{:}1 \end{array} \\ \begin{bmatrix} R^{2:2}_{2:2}\mathrm{id}_{\mathbb{I}[2,2]} & 0 & 0 \\ R^{1:2}_{2:2} f^{1:2}_{2:2} & R^{1:2}_{1:2}\mathrm{id}_{\mathbb{I}[1,2]} & 0 \\ 0 & R^{1:1}_{1:2} f^{1:1}_{1:2} & R^{1:1}_{1:1}\mathrm{id}_{\mathbb{I}[1,1]} \end{bmatrix} \end{array}.$$

Consequently, ΦR is

$$\begin{bmatrix} M_{2:2}^{2:2}R_{2:2}^{2:2}\mathrm{id}_{\mathbb{I}[2,2]} & 0 & 0 \\ (M_{2:2}^{1:2}R_{2:2}^{2:2} + M_{1:2}^{1:2}R_{2:2}^{1:2})f_{2:2}^{1:2} & M_{1:2}^{1:2}R_{1:2}^{1:2}\mathrm{id}_{\mathbb{I}[1,2]} & 0 \\ (M_{1:2}^{1:1}R_{2:2}^{1:2})(f_{1:2}^{1:1}f_{2:2}^{1:2}) & (M_{1:2}^{1:1}R_{1:2}^{1:2} + M_{1:1}^{1:1}R_{1:2}^{1:1})f_{1:2}^{1:1} & M_{1:1}^{1:1}R_{1:1}^{1:1}\mathrm{id}_{\mathbb{I}[1,1]} \end{bmatrix}.$$

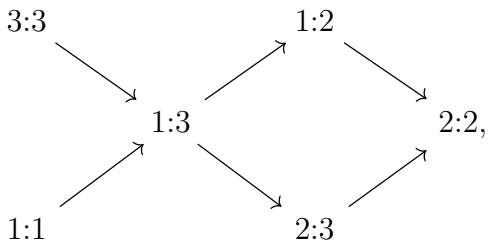
Since $f_{1:2}^{1:1}f_{2:2}^{1:2} = 0$, the lower left corner is still a zero block, as can be expected.

3.2. Algorithm

3.2.1. Input and Notation

Let τ_n be an orientation for $n \leq 4$ and M be a persistence module on $CL(\tau_n)$. In the previous section, we constructed the block matrix $\Phi(M) = \left[\Phi_{a:b}^{c:d}\right] = \left[M_{a:b}^{c:d}f_{a:b}^{c:d}\right]$ associated to M. Two blocks at distinct entries are said to be column (row) neighbors if they are in the same column (row). Neighbors need not be directly adjacent to one another in the block matrix. By an abuse of notation, we let $(c{:}d, a{:}b)$ referring to row $c{:}d$ and column $a{:}b$ also refer to the block (submatrix) located at that entry.

Recall that the vertices of the AR quiver $\Gamma(A_n(\tau_n))$ of $A_n(\tau_n)$ are in bijective correspondence to the interval representations. For that reason, the quiver structure of $\Gamma(A_n(\tau_n))$ naturally induces a partial order on the set of intervals by going from source vertices to sink vertices. A total order $\prec$ extending this can be fixed by resolving ambiguities using reverse lexicographic order on the pairs (b, d). For example, because $A_3(bf)$ has the following AR quiver



one obtains the order 3:3 $\prec$ 1:1 $\prec$ 1:3 $\prec$ 2:3 $\prec$ 1:2 $\prec$ 2:2. Here, the ambiguities are resolved as 3:3 $\prec$ 1:1, and 2:3 $\prec$ 1:2. We shall use $\prec$ to order the columns and rows of the block matrix.

Finally, we define the data that serve as input to Algorithm 1.

Definition 3.2.1. Let M be a persistence module on $CL(\tau_n)$. The *block matrix problem* of M is the block matrix $\Phi(M)$, together with permissible operations, and with rows and columns ordered as presented below.

1. If $\mathbb{I}[a, b] \trianglerighteq \mathbb{I}[c, d]$, then operations from row $a{:}b$ to row $c{:}d$ are permissible.

2. If $\mathbb{I}[a,b] \trianglerighteq \mathbb{I}[c,d]$, then operations to column $a{:}b$ from column $c{:}d$ are permissible.

Columns are ordered from left to right in increasing $\prec$, whereas rows from top to bottom are in decreasing $\prec$.

The permissible operations are the rules derived in Subsection 3.1.3. Applying these operations results in a block matrix isomorphic to $\Phi(M)$. For convenience, we distinguish the permissible operations that operate only within a fixed row or column block. An *inner row (column) operation* is a row (column) operation that only affects K-vector rows (columns) within some fixed row (column) $a{:}b$.

Recalling that if $\mathbb{I}[a,b] \not\trianglerighteq \mathbb{I}[c,d]$, then the block $(c{:}d, a{:}b)$ is always zero, even after the application of any permissible operation. To distinguish them from other blocks that are numerically zero, we designate them by $\emptyset$ and call these blocks *strongly zero blocks*.

Otherwise, for $\mathbb{I}[a,b] \trianglerighteq \mathbb{I}[c,d]$, we use the symbol $*$ to represent the block $M_{a:b}^{c:d}$ at $(c{:}d, a{:}b)$, which indicates that these blocks are so far *unprocessed*. As we operate on the block matrix, their status will be processed to either an identity matrix E or a zero matrix 0.

Notation 1. *To denote the possible block statuses, we use the following variables:*

- $*$ *for unprocessed blocks,*
- $\emptyset$ *for strongly zero blocks,*
- E *for identity blocks (of appropriate sizes), and*
- 0 *for zero blocks (of appropriate sizes).*

Blocks marked as $\emptyset$*,* E*, and* 0 *are considered processed.*

The block matrix might have numerically identity or zero blocks, although we label their status as being unprocessed $*$. This status merely reflects the fact that they have not yet been examined and fixed through the course of the algorithm.

Example 3.2.2. The block matrix problem corresponding to a persistence module on $CL_3(bf)$ has the form of

$$\begin{array}{c} \\ 2{:}2 \\ 1{:}2 \\ 2{:}3 \\ 1{:}3 \\ 1{:}1 \\ 3{:}3 \end{array} \begin{array}{c} \begin{array}{cccccc} 3{:}3 & 1{:}1 & 1{:}3 & 2{:}3 & 1{:}2 & 2{:}2 \end{array} \\ \begin{bmatrix} \emptyset & \emptyset & * & * & * & * \\ \emptyset & * & * & \emptyset & * & \emptyset \\ * & \emptyset & * & * & \emptyset & \emptyset \\ * & * & * & \emptyset & \emptyset & \emptyset \\ \emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset \\ * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \end{bmatrix} \end{array}.$$

3.2.2. Algorithm

Given a persistence module M on $CL(\tau_n)$ where $n \leq 4$, the input to Algorithm 1 is the block matrix problem of M. Below, we shall also use the notation M to denote the block matrix problem associated to the persistence module M.

The algorithm uses the following two facts. Given a usual K-matrix N, there exist invertible matrices R and S (of appropriate sizes) such that $RNS = \left[\begin{smallmatrix} E & 0 \\ 0 & 0 \end{smallmatrix}\right]$, a Smith normal form. Consequently, using appropriate inner row and column operations, a block $*$ can be transformed into the form $\left[\begin{smallmatrix} E & 0 \\ 0 & 0 \end{smallmatrix}\right]$. Meanwhile, using some identity submatrix E, a row (column) neighbor $*$ can be zeroed out using appropriate permissible column (row) operations. Complications arise from the side effects of these operations.

Algorithm 1 Main Algorithm. Input: a block matrix problem M

```
1:  procedure MATRIX_REDUCTION(M)
2:      while M has unprocessed submatrices do
3:          v_* ← the bottommost * block of the rightmost column with * blocks in M
4:          F_R ← ROW_SIDE_EFFECT(v_*)
5:          F_C ← COL_SIDE_EFFECT(v_*)
6:          Transform v_* to Smith normal form by inner operations on M.
7:          for all v' ∈ F_R do COL_FIX(v')
8:          for all v' ∈ F_C do ROW_FIX(v')
9:          Update the partitioning of blocks in block matrix M.
10:         while there exist nonzero blocks v_t with (p ← ERASABLE(v_t)) not null do
11:             Zero out v_t via the procedure indicated by p.
```

The main while loop of Algorithm 1 can be divided broadly into four main parts.

1. Transform one appropriate block v_* into a Smith normal form (Line 6) by inner row and column operations on M.
2. The operations performed in the previous part may affect the forms of neighboring identity blocks. We transform them back to identity blocks (Lines 7 and 8).
3. Update the partitioning of the blocks (Line 9). After obtaining the Smith normal form $\left[\begin{smallmatrix} E & 0 \\ 0 & 0 \end{smallmatrix}\right]$, we split up columns and rows so that each identity matrix E is its own block in M.
4. Greedily zero out erasable blocks v by addition of multiples of identity blocks.

We first illustrate parts one to three by an example based on the assumption that

$$M = \begin{bmatrix} * & * & v_* \\ * & \emptyset & E \end{bmatrix}$$

where operations from row 1 to row 2 and vice versa are impermissible. One obtains

$$M = \begin{bmatrix} * & * & v_* \\ * & \emptyset & E \end{bmatrix} \overset{1.}{\cong} \begin{bmatrix} * & * & \left[\begin{smallmatrix} E & 0 \\ 0 & 0 \end{smallmatrix}\right] \\ * & \emptyset & S \end{bmatrix} \overset{2.}{\cong} \begin{bmatrix} * & * & \left[\begin{smallmatrix} E & 0 \\ 0 & 0 \end{smallmatrix}\right] \\ * & \emptyset & E \end{bmatrix},$$

where the numbers above the isomorphisms represent the procedures being performed.

In the first part, the block v_* is chosen by the heuristic given in Algorithm 1 and Line 3. Therefore, v_* is dependent on the ordering of the rows and columns, which are fixed in Definition 3.2.1. By inner operations on M, the block v_* is transformed to Smith normal form. In particular, there are invertible matrices R and S such that $Rv_*S = \left[\begin{smallmatrix} E & 0 \\ 0 & 0 \end{smallmatrix}\right]$.

Algorithm 2

```
1: function COL_SIDE_EFFECT(v)
2:     return { v' | v' is an identity column neighbor of v }
1: function ROW_SIDE_EFFECT(v)
2:     return { v' | v' is an identity row neighbor of v }
```

Next, the block E below v_* becomes $ES = S$, possibly not an identity matrix. This change is recorded as a side effect. Since S is invertible, it can be transformed back using only inner row operations in ROW_FIX. In general there may be other identity blocks in the same row as S whose forms are affected by these row operations. To fix them, we recursively call ROW_FIX and COL_FIX in Algorithm 3. Checking that this does not engender an infinite recursion for the cases we consider is part of the proof of Theorem 3.2.3.

Algorithm 3

```
1: function COL_FIX(v)
2:     V' ← COL_SIDE_EFFECT(v)
3:     Transform v to an identity by inner column operations on M.
4:     for all v' ∈ V' do ROW_FIX(v')
1: function ROW_FIX(v)
2:     V' ← ROW_SIDE_EFFECT(v)
3:     Transform v to an identity by inner row operations on M.
4:     for all v' ∈ V' do COL_FIX(v')
```

In part three, the block matrix partitioning is updated to isolate the identity blocks E. Both the row and column of v_* are split into two. We obtain

$$M \cong \cdots \overset{3.}{=} \begin{bmatrix} * & * & E & 0 \\ * & * & 0 & 0 \\ * & \emptyset & E & 0 \\ * & \emptyset & 0 & E \end{bmatrix}.$$

Because v_* has a column neighbor E, the bottom row must also be split to isolate the parts of the old identity block.

Finally, we discuss part four. One simple case for a target block v_t to be erasable is when $v_t = (r, c)$ has a column neighbor identity block $v_E = (r', c)$ that has no nonzero row neighbors, and such that row operations from row r' to row r are permissible. Using

permissible row operations, the block v_t can be zeroed out by addition of a multiple of the identity block v_E. A similar statement holds if there exists a row neighbor identity block v_E satisfying similar conditions.

The cases above present no side effects. In general, zeroing out the target block v_t by addition of multiples of a row (column) may change the forms of other processed blocks. We separate the cases of row and column erasability in Algorithm 4.

Algorithm 4

```
1: function ERASABLE(v_t, v_f = null, visited = {})
2:     if ROW_ERASABLE(v_t, v_f, visited) is not null then
3:         return ROW_ERASABLE(v_t, v_f, visited)
4:     else if COL_ERASABLE(v_t, v_f, visited) is not null then
5:         return COL_ERASABLE(v_t, v_f, visited)
6:     else
7:         return null
```

In zeroing out the target v_t, we avoid changing the form of any previously obtained identity block. It is also possible that a zero block v_t' may become nonzero as a side effect. The algorithm ensures that if this occurs, then v_t' can and will be transformed back to 0 again. Iteratively, repairing these side effects may introduce more side effects. Therefore, we recursively call on our check for erasability on each side effect. To avoid any infinite recursion, we keep track of the targets v_t visited, and visit each block as a target at most once for each top-level call to ERASABLE.

If the conditions above can be satisfied, then the function ERASABLE returns a finite directed tree, called the *process tree*, which records the procedure to zero out v_t. Each vertex in a process tree is labeled with a pair (v_t, v_E) of a target block and an identity block that can be used to zero out v_t. The successor vertices (v_t', v_E') of a vertex (v_t, v_E) consist of all v_t' that appear as side effects in the operation to zero out v_t using v_E.

If no such procedure can be found, then ERASABLE returns a null (empty) process tree. This means that the block in question is declared as not being erasable in the current step of the algorithm.

Next, we discuss details of ROW_ERASABLE in Algorithm 5. In Line 2, the function COL_SIDE_EFFECT(v_t) is used to obtain candidate identity blocks v_E. We consider only unvisited blocks v_E where the row operation from v_E to v_t is permissible, and where v_E is not the flagged block v_f. Its purpose is clarified below.

Now, NONZERO_ROW_NEIGHBORS(v_E) is defined to return the set of row neighbors u of v_E that are neither zero nor strongly zero. Each u can potentially induce a side effect, which we check one by one. To illustrate, consider the following arrangement

$$\begin{array}{c} \\ r_1 \\ \\ r_2 \\ \end{array} \begin{bmatrix} \vdots & \vdots \\ \cdots\ u & \cdots v_E \cdots \\ \vdots & \vdots \\ \cdots\ v_t' & \cdots\ v_t \cdots \\ \vdots & \vdots \end{bmatrix}$$

Algorithm 5 Check whether or not v_t is row erasable without using block v_f.

1: **function** ROW_ERASABLE(v_t, v_f, *visited*)
2: $V' =$ COL_SIDE_EFFECT(v_t)
3: $visited \leftarrow visited \cup \{ v_t \}$
4: **for all** $v_E \in V'$ not in *visited*, $v_E \neq v_f$, and row operation from v_E to v_t permissible **do**
5: *usable* $\leftarrow$ true; *subtrees* $\leftarrow \{\}$
6: **for all** $u \in$ NONZERO_ROW_NEIGHBORS(v_E) **do**
7: $v_t' \leftarrow$ the block in same row as v_t and same column as u.
8: **if** v_t' is in *visited* or $v_t' = E$ **then**
9: *usable* $\leftarrow$ false; **break**
10: **if** ($v_t' = 0$ and ($p \leftarrow$ ERASABLE($v_t', u, visited$)) = null) **then**
11: *usable* $\leftarrow$ false; **break**
12: *subtrees* $\leftarrow$ *subtrees* $\cup \{p\}$.
13: **if** *usable* **then**
14: **return** process tree with root (v_t, v_E) and arrows to the roots of *subtrees*.
15: **return** null

where v_E is the identity block under consideration. Here, u is a nonzero row neighbor of v_E. Since we want to add multiples of row r_1 to r_2 to zero out v_t, the block v_t' in the same row r_2 as v_t and the same column as u (Line 7) might have its form affected.

The next few lines handle the checking of block v_t'. If the block v_t' is an identity block, or if it has already been visited previously, then we do not use row r_1. If the block v_t' is zero, we must check whether or not it can be transformed back to zero again. Here, the flag v_f comes into play. We set the flagged block as $v_f = u$ in the call to ERASABLE in Line 10, because we do not want to use u to zero out v_t', thereby undoing the operations to zero out v_t.

If a nonempty process tree is returned by the top-level call to ERASABLE(v_t) in Algorithm 1, then v_t is erasable. By construction, it suffices to traverse the process tree and do the operations indicated to zero out v_t and fix all side effects.

Let us state here the main theorem concerning Algorithm 1.

Theorem 3.2.3. Assume Algorithm 1 is called with the block matrix problem corresponding to a persistence module M on a commutative ladder of finite type. Then Algorithm 1 terminates in finite steps and the input matrix is transformed to an isomorphic block matrix consisting only of identity, zero, and strongly zero blocks, and whose indecomposable decomposition corresponds to an indecomposable decomposition of M.

Whether or not Algorithm 1 terminates depends not on the particular persistence module, but on the statuses of the blocks and the status changes brought about by the operations. Moreover, the operations to be performed depend only on the arrangement of the statuses. All these depend only on the initial arrangement, which in turn depends on the orientation τ_n and the ordering chosen for the intervals.

Algorithm 6 Check whether or not v_t is column erasable without using block v_f.

1: **function** COL_ERASABLE(v_t, v_f, *visited*)
2: $V' =$ ROW_SIDE_EFFECT(v_t)
3: $visited \leftarrow visited \cup \{ v_t \}$
4: **for all** $v_E \in V'$ not in *visited*, $v_E \neq v_f$, and column operation from v_E to v_t permissible **do**
5: *usable* $\leftarrow$ true; *subtrees* $\leftarrow \{\}$
6: **for all** $u \in$ NONZERO_COL_NEIGHBORS(v_E) **do**
7: $v_t' \leftarrow$ the block in same column as v_t and same row as u.
8: **if** v_t' is in *visited* or $v_t' = E$ **then**
9: *usable* $\leftarrow$ false; **break**
10: **if** ($v_t' = 0$ and ($p \leftarrow$ ERASABLE($v_t', u, visited$)) = null) **then**
11: *usable* $\leftarrow$ false; **break**
12: *subtrees* $\leftarrow$ *subtrees* $\cup \{p\}$.
13: **if** *usable* **then**
14: **return** process tree with root (v_t, v_E) and arrows to the roots of *subtrees*.
15: **return** null

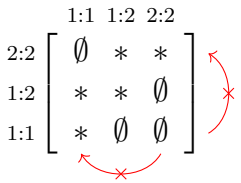
From a result in [16], a commutative ladder $CL(\tau_n)$ is finite type if and only if $n \leq 4$, so that there are only a finite number of cases to check. Below, we provide proofs for Theorem 3.2.3 with orientations b, bf, and bfb. The proofs for the other orientations are similar.

Furthermore, an indecomposable decomposition can easily be read off the resulting normal form consisting of only identity, zero, and strongly zero block. Correspondence to an indecomposable decomposition of the persistence module M is provided by Theorem 3.1.1.

We were unable to find a proof that does not involve manual checking of each possible orientation. Given a particular persistence module, it is clear that for each completed iteration of the main while loop in Algorithm 1, the total number of scalar entries in unprocessed blocks strictly decreases. Moreover, the procedure ERASABLE avoids any infinite recursion by construction. The difficulty comes from the use of Algorithm 1, Line 3 for choosing v_* and subsequently showing that all side effects can always be resolved.

Case $CL_2(b)$

The input block matrix problem is generally of the following form.



Initially, all top to bottom and left to right operations are impermissible. The red arrows indicate the additional impermissible operations.

First the unprocessed block at $v_* = (2{:}2, 2{:}2)$ is transformed by inner elementary operations to Smith normal form $\left[\begin{smallmatrix} E & 0 \\ 0 & 0 \end{smallmatrix}\right]$. Furthermore, v_* has no identity neighbor: no side effects exist to undo.

Updating the block partitioning, the matrix is now in the form

$$\begin{array}{c} \\ 2{:}2_1 \\ 2{:}2_2 \\ 1{:}2 \\ 1{:}1 \end{array} \begin{array}{c} \begin{array}{cccc} 1{:}1 & 1{:}2 & 2{:}2_1 & 2{:}2_2 \end{array} \\ \begin{bmatrix} \emptyset & * & E & 0 \\ \emptyset & * & 0 & 0 \\ * & * & \emptyset & \emptyset \\ * & \emptyset & \emptyset & \emptyset \end{bmatrix} \end{array}.$$

For convenience, we use subscripts to distinguish the two columns and rows corresponding to 2:2 obtained after the repartitioning. Additions from the columns in $2{:}2_1$ to the columns in 1:2 are permitted. The unprocessed submatrix $(2{:}2_1, 1{:}2)$ is erasable using the newly processed E, with no side effects. Thereby, one obtains the form

$$\begin{array}{c} \\ 2{:}2_1 \\ 2{:}2_2 \\ 1{:}2 \\ 1{:}1 \end{array} \begin{array}{c} \begin{array}{cccc} 1{:}1 & 1{:}2 & 2{:}2_1 & 2{:}2_2 \end{array} \\ \begin{bmatrix} \emptyset & 0 & E & 0 \\ \emptyset & * & 0 & 0 \\ * & * & \emptyset & \emptyset \\ * & \emptyset & \emptyset & \emptyset \end{bmatrix} \end{array} \cong \begin{array}{c} \\ 2{:}2_2 \\ 1{:}2 \\ 1{:}1 \end{array} \begin{array}{c} \begin{array}{cc} 1{:}1 & 1{:}2 \end{array} \\ \begin{bmatrix} \emptyset & * \\ * & * \\ * & \emptyset \end{bmatrix} \end{array} \bigoplus \begin{array}{c} \\ 2{:}2_1 \end{array} \begin{array}{c} \begin{array}{c} 2{:}2_1 \end{array} \\ \begin{bmatrix} E \end{bmatrix} \end{array} \bigoplus \begin{array}{c} \\ \end{array} \begin{array}{c} \begin{array}{c} 2{:}2_2 \end{array} \\ \begin{bmatrix} \ \end{bmatrix} \end{array} \tag{3.5}$$

which we have expressed as a direct sum of block matrices.

Here, two indecomposable representations of $CL_2(b)$ can be extracted. The identity submatrix E in $(2{:}2_1, 2{:}2_1)$ is

$$Ef_{2:2}^{2:2} = \begin{bmatrix} 1f_{2:2}^{2:2} & 0 & \dots & 0 \\ 0 & 1f_{2:2}^{2:2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1f_{2:2}^{2:2} \end{bmatrix}$$

where

$$f_{2:2}^{2:2} = \begin{array}{ccc} 0 & \longleftarrow & K \\ \uparrow & & \uparrow{\scriptstyle \mathrm{id}_K} \\ 0 & \longleftarrow & K \end{array}, \tag{3.6}$$

as in the proof of Lemma 3.1.3.

Via the isomorphism functor F in Theorem 3.1.1, the arrow $f_{2:2}^{2:2}$ in Eq. (3.6) can be regarded as the corresponding representation $F^{-1}(f_{2:2}^{2:2})$. This is indecomposable. Consequently, $Ef_{2:2}^{2:2}$ corresponds to a direct sum of m copies of the representation in Eq. (3.6), where m represents the size of E.

The third term in Eq. (3.5) is an empty matrix with 0 rows, and represents the arrow $0 : \mathbb{I}[2,2]^{m_1} \to 0$ in $\operatorname{rep} A_2(b)$, where m_1 represents the number of K-vector columns

in $2{:}2_2$. By the isomorphism, this corresponds to a direct sum of m_1 copies of the indecomposable representation:

$$\begin{array}{ccc} 0 & \longleftarrow & 0 \\ \uparrow & & \uparrow \\ 0 & \longleftarrow & K \end{array}.$$

Now, the row $2{:}2_1$ and columns $2{:}2_1, 2{:}2_2$ in the block matrix Eq. (3.5) will neither affect nor be affected by subsequent operations. Therefore, we hide them from the block matrix. The unprocessed block $(1{:}2, 1{:}2)$ is next transformed to Smith normal form to obtain the following.

$$\begin{array}{c} \\ 2{:}2 \\ 1{:}2 \\ 1{:}1 \end{array}\!\!\begin{array}{c} \begin{array}{cc} 1{:}1 & 1{:}2 \end{array} \\ \begin{bmatrix} \emptyset & * \\ * & * \\ * & \emptyset \end{bmatrix} \end{array} \cong \begin{array}{c} \\ 2{:}2 \\ 1{:}2_1 \\ 1{:}2_2 \\ 1{:}1 \end{array}\!\!\begin{array}{c} \begin{array}{ccc} 1{:}1 & 1{:}2_1 & 1{:}2_2 \end{array} \\ \begin{bmatrix} \emptyset & * & * \\ * & E & 0 \\ * & 0 & 0 \\ * & \emptyset & \emptyset \end{bmatrix} \end{array}.$$

We see that $(2{:}2, 1{:}2_1)$ is erasable using $(1{:}2_1, 1{:}2_1)$. The checking via ROW_ERASABLE in Algorithm 5 proceeds as follows. Whereas $u = (1{:}2_1, 1{:}1)$ is a nonzero row neighbor of E, the computed potential side effect is $v'_t = (2{:}2, 1{:}1)$. Since v'_t is strongly zero, addition from row $1{:}2_1$ will not affect it.

Similarly, $(1{:}2_1, 1{:}1)$ is erasable. After zeroing out erasable blocks, one obtains

$$\begin{array}{c} \\ 2{:}2 \\ 1{:}2_1 \\ 1{:}2_2 \\ 1{:}1 \end{array}\!\!\begin{array}{c} \begin{array}{ccc} 1{:}1 & 1{:}2_1 & 1{:}2_2 \end{array} \\ \begin{bmatrix} \emptyset & 0 & * \\ * & E & 0 \\ * & 0 & 0 \\ * & \emptyset & \emptyset \end{bmatrix} \end{array} \text{ and then } \begin{array}{c} \\ 2{:}2 \\ 1{:}2_1 \\ 1{:}2_2 \\ 1{:}1 \end{array}\!\!\begin{array}{c} \begin{array}{ccc} 1{:}1 & 1{:}2_1 & 1{:}2_2 \end{array} \\ \begin{bmatrix} \emptyset & 0 & * \\ 0 & E & 0 \\ * & 0 & 0 \\ * & \emptyset & \emptyset \end{bmatrix} \end{array}.$$

The identity submatrix E in $(1{:}2_1, 1{:}2_1)$ corresponds to copies of the indecomposable representation

$$\begin{array}{ccc} K & \longleftarrow & K \\ \uparrow & & \uparrow \\ K & \longleftarrow & K \end{array}$$

as direct summands.

Once again we abbreviate the block matrix as

$$\begin{array}{c} \\ 2{:}2 \\ 1{:}2 \\ 1{:}1 \end{array}\!\!\begin{array}{c} \begin{array}{cc} 1{:}1 & 1{:}2 \end{array} \\ \begin{bmatrix} \emptyset & * \\ * & 0 \\ * & \emptyset \end{bmatrix} \end{array} \text{ and then } \begin{array}{c} \\ 2{:}2_1 \\ 2{:}2_2 \\ 1{:}2 \\ 1{:}1 \end{array}\!\!\begin{array}{c} \begin{array}{ccc} 1{:}1 & 1{:}2_1 & 1{:}2_2 \end{array} \\ \begin{bmatrix} \emptyset & E & 0 \\ \emptyset & 0 & 0 \\ * & 0 & 0 \\ * & \emptyset & \emptyset \end{bmatrix} \end{array}$$

after transforming the next target $(2{:}2, 1{:}2)$ to Smith normal form. The identity submatrix in $(2{:}2_1, 1{:}2_1)$, the row $2{:}2_2$, and the column $1{:}2_2$ respectively correspond to copies

of the indecomposable representations with dimension vectors $\begin{smallmatrix}01\\11\end{smallmatrix}$, $\begin{smallmatrix}01\\00\end{smallmatrix}$, and $\begin{smallmatrix}00\\11\end{smallmatrix}$ as direct summands.

What remains is the form $\begin{smallmatrix}&1{:}1\\1{:}2\\1{:}1\end{smallmatrix}\begin{bmatrix}*\\*\end{bmatrix}$, from which one obtains

$$\begin{smallmatrix}&&1{:}1_1&1{:}1_2\\1{:}2\\1{:}1_1\\1{:}1_2\end{smallmatrix}\begin{bmatrix}0&*\\E&0\\0&0\end{bmatrix}$$

after transforming the next target $(1{:}1, 1{:}1)$ to normal form, and zeroing out erasable blocks. The identity submatrix $(1{:}1_1, 1{:}1_1)$ and the row $1{:}1_2$ respectively correspond to copies of the indecomposable representations with dimension vectors $\begin{smallmatrix}10\\10\end{smallmatrix}$ and $\begin{smallmatrix}10\\00\end{smallmatrix}$.

Abbreviating again, we are left with $\begin{smallmatrix}&1{:}1\\1{:}2\end{smallmatrix}\begin{bmatrix}*\end{bmatrix}$. Transforming the last target $(1{:}2, 1{:}1)$ to normal form yields

$$\begin{smallmatrix}&&1{:}1_1&1{:}1_2\\1{:}2_1\\1{:}2_2\end{smallmatrix}\begin{bmatrix}E&0\\0&0\end{bmatrix}.$$

The identity submatrix $(1{:}2_1, 1{:}1_1)$, the row $1{:}2_2$, and the column $1{:}1_2$ respectively correspond to copies of the indecomposable representations with dimension vectors $\begin{smallmatrix}11\\10\end{smallmatrix}$, $\begin{smallmatrix}11\\00\end{smallmatrix}$, and $\begin{smallmatrix}00\\10\end{smallmatrix}$.

By this point, all possible indecomposable representations of $CL_2(b)$ have been obtained. This result can be confirmed, for example, by checking with the Auslander–Reiten quiver of $CL_2(b)$. Given a particular persistence module M on $CL_2(b)$, the algorithm gives the multiplicities of each of these indecomposables in an indecomposable decomposition of M.

Case $CL_3(bf)$

The input block matrix is given in Example 3.2.2.

Although the presence of impermissible operations caused no noticeable complications in the case of $CL_2(b)$, in general this is not so. For better readability, only the relevant impermissible operations at each step are included below. Each numbered step corresponds to one pass of the outer while loop in Algorithm 1.

1. At the first loop, v_* is $(2{:}2, 2{:}2)$.

a) Transform $(2{:}2, 2{:}2)$ to Smith normal form, giving the block matrix

$$\begin{array}{c} \\ 2{:}2_1 \\ 2{:}2_2 \\ 1{:}2 \\ 2{:}3 \\ 1{:}3 \\ 1{:}1 \\ 3{:}3 \end{array} \begin{array}{c} \begin{array}{ccccccc} 3{:}3 & 1{:}1 & 1{:}3 & 2{:}3 & 1{:}2 & 2{:}2_1 & 2{:}2_2 \end{array} \\ \begin{bmatrix} \emptyset & \emptyset & * & * & * & E & 0 \\ \emptyset & \emptyset & * & * & * & 0 & 0 \\ \emptyset & * & * & \emptyset & * & \emptyset & \emptyset \\ * & \emptyset & * & * & \emptyset & \emptyset & \emptyset \\ * & * & * & \emptyset & \emptyset & \emptyset & \emptyset \\ \emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \\ * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \end{bmatrix} \end{array}.$$

b) Zero out the blocks $(2{:}2_1, 1{:}3)$, $(2{:}2_1, 2{:}3)$, and $(2{:}2_1, 1{:}2)$ by the identity submatrix at $(2{:}2_1, 2{:}2_1)$:

$$\begin{array}{c} \\ 2{:}2_1 \\ 2{:}2_2 \\ 1{:}2 \\ 2{:}3 \\ 1{:}3 \\ 1{:}1 \\ 3{:}3 \end{array} \begin{array}{c} \begin{array}{ccccccc} 3{:}3 & 1{:}1 & 1{:}3 & 2{:}3 & 1{:}2 & 2{:}2_1 & 2{:}2_2 \end{array} \\ \begin{bmatrix} \emptyset & \emptyset & 0 & 0 & 0 & E & 0 \\ \emptyset & \emptyset & * & * & * & 0 & 0 \\ \emptyset & * & * & \emptyset & * & \emptyset & \emptyset \\ * & \emptyset & * & * & \emptyset & \emptyset & \emptyset \\ * & * & * & \emptyset & \emptyset & \emptyset & \emptyset \\ \emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \\ * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \end{bmatrix} \end{array}.$$

c) The identity submatrix $(2{:}2_1, 2{:}2_1)$ and the columns in $2{:}2_2$ give copies of indecomposable representations isomorphic to

$$\begin{array}{ccccc} 0 & \longleftarrow & K & \longrightarrow & 0 \\ \uparrow & & \uparrow & & \uparrow \\ 0 & \longleftarrow & K & \longrightarrow & 0 \end{array} \quad \text{and} \quad \begin{array}{ccccc} 0 & \longleftarrow & 0 & \longrightarrow & 0 \\ \uparrow & & \uparrow & & \uparrow \\ 0 & \longleftarrow & K & \longrightarrow & 0 \end{array}$$

corresponding to the vertices $\begin{smallmatrix} 010 \\ 010 \end{smallmatrix}$ and $\begin{smallmatrix} 000 \\ 010 \end{smallmatrix}$ in Figure 2.1.

d) We are left with

$$\begin{array}{c} \\ 2{:}2 \\ 1{:}2 \\ 2{:}3 \\ 1{:}3 \\ 1{:}1 \\ 3{:}3 \end{array} \begin{array}{c} \begin{array}{ccccc} 3{:}3 & 1{:}1 & 1{:}3 & 2{:}3 & 1{:}2 \end{array} \\ \begin{bmatrix} \emptyset & \emptyset & * & * & * \\ \emptyset & * & * & \emptyset & * \\ * & \emptyset & * & * & \emptyset \\ * & * & * & \emptyset & \emptyset \\ \emptyset & * & \emptyset & \emptyset & \emptyset \\ * & \emptyset & \emptyset & \emptyset & \emptyset \end{bmatrix} \end{array}.$$

2. Next, v_* is $(1{:}2, 1{:}2)$. The direct summands with dimension vector $\begin{smallmatrix} 110 \\ 110 \end{smallmatrix}$ can be extracted

in the same way. The block matrix is now

$$\begin{array}{c} \\ 2{:}2 \\ 1{:}2 \\ 2{:}3 \\ 1{:}3 \\ 1{:}1 \\ 3{:}3 \end{array}\!\!\begin{array}{c} \begin{array}{ccccc} 3{:}3 & 1{:}1 & 1{:}3 & 2{:}3 & 1{:}2 \end{array} \\ \begin{bmatrix} \emptyset & \emptyset & * & * & * \\ \emptyset & * & * & \emptyset & 0 \\ * & \emptyset & * & * & \emptyset \\ * & * & * & \emptyset & \emptyset \\ \emptyset & * & \emptyset & \emptyset & \emptyset \\ * & \emptyset & \emptyset & \emptyset & \emptyset \end{bmatrix} \end{array}.$$

3. Here, v_* is $(2{:}2, 1{:}2)$. After transforming v_* to Smith normal form, the block matrix is

$$\begin{array}{c} \\ 2{:}2_1 \\ 2{:}2_2 \\ 1{:}2 \\ 2{:}3 \\ 1{:}3 \\ 1{:}1 \\ 3{:}3 \end{array}\!\!\begin{array}{c} \begin{array}{cccccc} 3{:}3 & 1{:}1 & 1{:}3 & 2{:}3 & 1{:}2_1 & 1{:}2_2 \end{array} \\ \begin{bmatrix} \emptyset & \emptyset & * & * & E & 0 \\ \emptyset & \emptyset & * & * & 0 & 0 \\ \emptyset & * & * & \emptyset & 0 & 0 \\ * & \emptyset & * & * & \emptyset & \emptyset \\ * & * & * & \emptyset & \emptyset & \emptyset \\ \emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset \\ * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \end{bmatrix} \end{array} \text{ and then } \begin{array}{c} \\ 2{:}2_1 \\ 2{:}2_2 \\ 1{:}2 \\ 2{:}3 \\ 1{:}3 \\ 1{:}1 \\ 3{:}3 \end{array}\!\!\begin{array}{c} \begin{array}{cccccc} 3{:}3 & 1{:}1 & 1{:}3 & 2{:}3 & 1{:}2_1 & 1{:}2_2 \end{array} \\ \begin{bmatrix} \emptyset & \emptyset & 0 & * & E & 0 \\ \emptyset & \emptyset & * & * & 0 & 0 \\ \emptyset & * & * & \emptyset & 0 & 0 \\ * & \emptyset & * & * & \emptyset & \emptyset \\ * & * & * & \emptyset & \emptyset & \emptyset \\ \emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset \\ * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \end{bmatrix} \end{array}.$$

Whereas the block $(2{:}2_1, 1{:}3)$ is erasable, $(2{:}2_1, 2{:}3)$ is not erasable so far, because column operations from $1{:}2_1$ to $2{:}3$ is impermissible. Direct summands with the dimension vector $\begin{smallmatrix}000\\110\end{smallmatrix}$ can be extracted, and we are left with

$$\begin{array}{c} \\ 2{:}2_1 \\ 2{:}2_2 \\ 1{:}2 \\ 2{:}3 \\ 1{:}3 \\ 1{:}1 \\ 3{:}3 \end{array}\!\!\begin{array}{c} \begin{array}{ccccc} 3{:}3 & 1{:}1 & 1{:}3 & 2{:}3 & 1{:}2 \end{array} \\ \begin{bmatrix} \emptyset & \emptyset & 0 & * & E \\ \emptyset & \emptyset & * & * & 0 \\ \emptyset & * & * & \emptyset & 0 \\ * & \emptyset & * & * & \emptyset \\ * & * & * & \emptyset & \emptyset \\ \emptyset & * & \emptyset & \emptyset & \emptyset \\ * & \emptyset & \emptyset & \emptyset & \emptyset \end{bmatrix} \end{array}.$$

4. Transforming $v_* = (2{:}3, 2{:}3)$, we obtain

$$\begin{array}{c} \\ 2{:}2_1 \\ 2{:}2_2 \\ 1{:}2 \\ 2{:}3_1 \\ 2{:}3_2 \\ 1{:}3 \\ 1{:}1 \\ 3{:}3 \end{array}\!\!\begin{array}{c} \begin{array}{cccccc} 3{:}3 & 1{:}1 & 1{:}3 & 2{:}3_1 & 2{:}3_2 & 1{:}2 \end{array} \\ \begin{bmatrix} \emptyset & \emptyset & 0 & * & * & E \\ \emptyset & \emptyset & * & * & * & 0 \\ \emptyset & * & * & \emptyset & \emptyset & 0 \\ * & \emptyset & * & E & 0 & \emptyset \\ * & \emptyset & * & 0 & 0 & \emptyset \\ * & * & * & \emptyset & \emptyset & \emptyset \\ \emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset \\ * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \end{bmatrix} \end{array}.$$

It is readily apparent that the blocks $(2{:}2_2, 2{:}3_1)$ and $(2{:}3_1, 3{:}3)$ are erasable by the identity $(2{:}3_1, 2{:}3_1)$ without any concerns of side effects. Moreover at this point, the recursive checking of erasability in Algorithm 5 and 6 works well; the block $v_t = (2{:}2_1, 2{:}3_1)$ is also erasable via the following procedures. Zeroing v_t out by additions from the identity $(2{:}3_1, 2{:}3_1)$, the processed block $(2{:}2_1, 1{:}3)$ may become nonzero by side effect with $(2{:}3_1, 1{:}3)$. However, $(2{:}2_1, 1{:}3)$ can be zeroed out again by additions from the identity $(2{:}2_1, 1{:}2)$. Similarly, the block $(2{:}3_1, 1{:}3)$ is also erasable. We thus extract the direct summands with the dimension vector $\begin{smallmatrix}011\\011\end{smallmatrix}$, and have the block matrix

$$\begin{array}{c} \\ 2{:}2_1 \\ 2{:}2_2 \\ 1{:}2 \\ 2{:}3 \\ 1{:}3 \\ 1{:}1 \\ 3{:}3 \end{array} \begin{array}{c} \begin{array}{ccccc} 3{:}3 & 1{:}1 & 1{:}3 & 2{:}3 & 1{:}2 \end{array} \\ \begin{bmatrix} \emptyset & \emptyset & 0 & * & E \\ \emptyset & \emptyset & * & * & 0 \\ \emptyset & * & * & \emptyset & 0 \\ * & \emptyset & * & 0 & \emptyset \\ * & * & * & \emptyset & \emptyset \\ \emptyset & * & \emptyset & \emptyset & \emptyset \\ * & \emptyset & \emptyset & \emptyset & \emptyset \end{bmatrix} \end{array}.$$

5. v_* is $(2{:}2_2, 2{:}3)$, and in the same manner, direct summands $\begin{smallmatrix}010\\011\end{smallmatrix}$ can be extracted.

$$\begin{array}{c} \\ 2{:}2_1 \\ 2{:}2_2 \\ 1{:}2 \\ 2{:}3 \\ 1{:}3 \\ 1{:}1 \\ 3{:}3 \end{array} \begin{array}{c} \begin{array}{ccccc} 3{:}3 & 1{:}1 & 1{:}3 & 2{:}3 & 1{:}2 \end{array} \\ \begin{bmatrix} \emptyset & \emptyset & 0 & * & E \\ \emptyset & \emptyset & * & 0 & 0 \\ \emptyset & * & * & \emptyset & 0 \\ * & \emptyset & * & 0 & \emptyset \\ * & * & * & \emptyset & \emptyset \\ \emptyset & * & \emptyset & \emptyset & \emptyset \\ * & \emptyset & \emptyset & \emptyset & \emptyset \end{bmatrix} \end{array}$$

6. Transforming $v_* = (2{:}2_1, 2{:}3)$, then the row neighbor $(2{:}2_1, 1{:}2)$ may not be the identity. The function COL_FIX is called to transform it to an identity again, then we obtain:

$$\begin{array}{c} \\ 2{:}2_1 \\ 2{:}2_2 \\ 2{:}2_3 \\ 1{:}2 \\ 2{:}3 \\ 1{:}3 \\ 1{:}1 \\ 3{:}3 \end{array} \begin{array}{c} \begin{array}{ccccccc} 3{:}3 & 1{:}1 & 1{:}3 & 2{:}3_1 & 2{:}3_2 & 1{:}2_1 & 1{:}2_2 \end{array} \\ \begin{bmatrix} \emptyset & \emptyset & 0 & E & 0 & E & 0 \\ \emptyset & \emptyset & 0 & 0 & 0 & 0 & E \\ \emptyset & \emptyset & * & 0 & 0 & 0 & 0 \\ \emptyset & * & * & \emptyset & \emptyset & 0 & 0 \\ * & \emptyset & * & 0 & 0 & \emptyset & \emptyset \\ * & * & * & \emptyset & \emptyset & \emptyset & \emptyset \\ \emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \\ * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \end{bmatrix} \end{array}.$$

The block matrix has the following decomposition

$$\begin{array}{c} \\ 2{:}2_1 \\ 2{:}2_2 \\ 2{:}2_3 \\ 1{:}2 \\ 2{:}3 \\ 1{:}3 \\ 1{:}1 \\ 3{:}3 \end{array}\!\!\begin{array}{c} \begin{array}{ccccccc} 3{:}3 & 1{:}1 & 1{:}3 & 2{:}3_1 & 2{:}3_2 & 1{:}2_1 & 1{:}2_2 \end{array} \\ \begin{bmatrix} \emptyset & \emptyset & 0 & E & 0 & E & 0 \\ \emptyset & \emptyset & 0 & 0 & 0 & 0 & E \\ \emptyset & \emptyset & * & 0 & 0 & 0 & 0 \\ \emptyset & * & * & \emptyset & \emptyset & 0 & 0 \\ * & \emptyset & * & 0 & 0 & \emptyset & \emptyset \\ * & * & * & \emptyset & \emptyset & \emptyset & \emptyset \\ \emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \\ * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \end{bmatrix} \end{array} \cong \begin{array}{c} \\ 2{:}2_3 \\ 1{:}2 \\ 2{:}3 \\ 1{:}3 \\ 1{:}1 \\ 3{:}3 \end{array}\!\!\begin{array}{c} \begin{array}{ccc} 3{:}3 & 1{:}1 & 1{:}3 \end{array} \\ \begin{bmatrix} \emptyset & \emptyset & * \\ \emptyset & * & * \\ * & \emptyset & * \\ * & * & * \\ \emptyset & * & \emptyset \\ * & \emptyset & \emptyset \end{bmatrix} \end{array}$$

$$\bigoplus_{2:2_1} \begin{array}{c} \begin{array}{cc} 2{:}3_1 & 1{:}2_1 \end{array} \\ \begin{bmatrix} E & E \end{bmatrix} \end{array} \bigoplus_{2:2_2} \begin{array}{c} 1{:}2_2 \\ \begin{bmatrix} E \end{bmatrix} \end{array} \bigoplus \begin{array}{c} 2{:}3_2 \\ [\quad] \end{array}.$$

The extracted third and fourth terms are $\begin{smallmatrix} 010 \\ 110 \end{smallmatrix}$ and $\begin{smallmatrix} 000 \\ 011 \end{smallmatrix}$. The second direct summand is copies of the arrow $\begin{bmatrix} f_{2:3}^{2:2} & f_{1:2}^{2:2} \end{bmatrix} : \mathbb{I}[2,3] \oplus \mathbb{I}[1,2] \to \mathbb{I}[2,2]$. Through the isomorphism in Theorem 3.1.1, each arrow is isomorphic to the indecomposable representation with the dimension vector $\begin{smallmatrix} 010 \\ 121 \end{smallmatrix}$ given in Eq. (2.4).

7. The procedures from this step are executed in a similar way we have done, and one obtains direct summands with the dimension vectors $\begin{smallmatrix} 111 \\ 111 \end{smallmatrix}$, $\begin{smallmatrix} 110 \\ 111 \end{smallmatrix}$, $\begin{smallmatrix} 010 \\ 111 \end{smallmatrix}$, $\begin{smallmatrix} 000 \\ 111 \end{smallmatrix}$, $\begin{smallmatrix} 010 \\ 000 \end{smallmatrix}$, $\begin{smallmatrix} 121 \\ 111 \end{smallmatrix}$, $\begin{smallmatrix} 011 \\ 111 \end{smallmatrix}$, $\begin{smallmatrix} 100 \\ 100 \end{smallmatrix}$, $\begin{smallmatrix} 100 \\ 000 \end{smallmatrix}$, $\begin{smallmatrix} 110 \\ 100 \end{smallmatrix}$, $\begin{smallmatrix} 000 \\ 100 \end{smallmatrix}$, $\begin{smallmatrix} 110 \\ 000 \end{smallmatrix}$, $\begin{smallmatrix} 001 \\ 001 \end{smallmatrix}$, $\begin{smallmatrix} 001 \\ 000 \end{smallmatrix}$, $\begin{smallmatrix} 111 \\ 001 \end{smallmatrix}$, $\begin{smallmatrix} 111 \\ 000 \end{smallmatrix}$, $\begin{smallmatrix} 111 \\ 101 \end{smallmatrix}$, $\begin{smallmatrix} 111 \\ 100 \end{smallmatrix}$, $\begin{smallmatrix} 011 \\ 001 \end{smallmatrix}$, $\begin{smallmatrix} 011 \\ 000 \end{smallmatrix}$, and $\begin{smallmatrix} 000 \\ 001 \end{smallmatrix}$.

Case $CL_4(bfb)$

An erasable identity block considered in Algorithm 4 has not appeared yet. As an example of this issue, let us give a quick proof for $CL_4(bfb)$, where such an issue arises during the procedures on an unprocessed block in the column 2:3.

The input block matrix is generally of the form

$$\begin{array}{c} \\ 2{:}2 \\ 4{:}4 \\ 1{:}2 \\ 2{:}4 \\ 1{:}4 \\ 2{:}3 \\ 1{:}3 \\ 3{:}4 \\ 1{:}1 \\ 3{:}3 \end{array}\!\!\begin{array}{c} \begin{array}{cccccccccc} 3{:}3 & 1{:}1 & 3{:}4 & 1{:}3 & 2{:}3 & 1{:}4 & 2{:}4 & 1{:}2 & 4{:}4 & 2{:}2 \end{array} \\ \begin{bmatrix} \emptyset & \emptyset & \emptyset & * & * & * & * & * & \emptyset & * \\ \emptyset & \emptyset & * & \emptyset & \emptyset & * & * & \emptyset & * & \emptyset \\ \emptyset & * & \emptyset & * & \emptyset & * & \emptyset & * & \emptyset & \emptyset \\ * & \emptyset & * & * & * & * & * & \emptyset & \emptyset & \emptyset \\ * & * & * & * & \emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset \\ * & \emptyset & \emptyset & * & * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \\ * & * & \emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \\ * & \emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \\ \emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \\ * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \end{bmatrix} \end{array}.$$

For the steps similar to those already done in earlier cases, we only provide the block matrix form after sequences of operations for each column b:d.

1. Procedures on column 2:2 yield direct summands with the dimension vectors $\begin{smallmatrix}0100\\0100\end{smallmatrix}$ and $\begin{smallmatrix}0000\\0100\end{smallmatrix}$, leaving the block matrix form presented below.

$$\begin{array}{c} \\ 2{:}2 \\ 4{:}4 \\ 1{:}2 \\ 2{:}4 \\ 1{:}4 \\ 2{:}3 \\ 1{:}3 \\ 3{:}4 \\ 1{:}1 \\ 3{:}3 \end{array} \begin{array}{c} \begin{array}{ccccccccc} 3{:}3 & 1{:}1 & 3{:}4 & 1{:}3 & 2{:}3 & 1{:}4 & 2{:}4 & 1{:}2 & 4{:}4 \end{array} \\ \left[\begin{array}{ccccccccc} \emptyset & \emptyset & \emptyset & * & * & * & * & * & \emptyset \\ \emptyset & \emptyset & * & \emptyset & \emptyset & * & * & \emptyset & * \\ \emptyset & * & \emptyset & * & \emptyset & * & \emptyset & * & \emptyset \\ * & \emptyset & * & * & * & * & * & \emptyset & \emptyset \\ * & * & * & * & \emptyset & * & \emptyset & \emptyset & \emptyset \\ * & \emptyset & \emptyset & * & * & \emptyset & \emptyset & \emptyset & \emptyset \\ * & * & \emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \\ * & \emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \\ \emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \\ * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \end{array}\right] \end{array}$$

2. Next is the column 4:4, extracting $\begin{smallmatrix}0001\\0001\end{smallmatrix}$ and $\begin{smallmatrix}0000\\0001\end{smallmatrix}$ and yielding the following form.

$$\begin{array}{c} \\ 2{:}2 \\ 4{:}4 \\ 1{:}2 \\ 2{:}4 \\ 1{:}4 \\ 2{:}3 \\ 1{:}3 \\ 3{:}4 \\ 1{:}1 \\ 3{:}3 \end{array} \begin{array}{c} \begin{array}{cccccccc} 3{:}3 & 1{:}1 & 3{:}4 & 1{:}3 & 2{:}3 & 1{:}4 & 2{:}4 & 1{:}2 \end{array} \\ \left[\begin{array}{cccccccc} \emptyset & \emptyset & \emptyset & * & * & * & * & * \\ \emptyset & \emptyset & * & \emptyset & \emptyset & * & * & \emptyset \\ \emptyset & * & \emptyset & * & \emptyset & * & \emptyset & * \\ * & \emptyset & * & * & * & * & * & \emptyset \\ * & * & * & * & \emptyset & * & \emptyset & \emptyset \\ * & \emptyset & \emptyset & * & * & \emptyset & \emptyset & \emptyset \\ * & * & \emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset \\ * & \emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \\ \emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \\ * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \end{array}\right] \end{array}$$

3. The subsequent column is 1:2, extracting $\begin{smallmatrix}1100\\1100\end{smallmatrix}$ and $\begin{smallmatrix}0000\\1100\end{smallmatrix}$. One obtains the block matrix

$$\begin{array}{c} \\ 2{:}2_1 \\ 2{:}2_2 \\ 4{:}4 \\ 1{:}2 \\ 2{:}4 \\ 1{:}4 \\ 2{:}3 \\ 1{:}3 \\ 3{:}4 \\ 1{:}1 \\ 3{:}3 \end{array} \begin{array}{c} \begin{array}{cccccccc} 3{:}3 & 1{:}1 & 3{:}4 & 1{:}3 & 2{:}3 & 1{:}4 & 2{:}4 & 1{:}2 \end{array} \\ \left[\begin{array}{cccccccc} \emptyset & \emptyset & \emptyset & 0 & * & 0 & * & E \\ \emptyset & \emptyset & \emptyset & * & * & * & * & 0 \\ \emptyset & \emptyset & * & \emptyset & \emptyset & * & * & \emptyset \\ \emptyset & * & \emptyset & * & \emptyset & * & \emptyset & 0 \\ * & \emptyset & * & * & * & * & * & \emptyset \\ * & * & * & * & \emptyset & * & \emptyset & \emptyset \\ * & \emptyset & \emptyset & * & * & \emptyset & \emptyset & \emptyset \\ * & * & \emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset \\ * & \emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \\ \emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \\ * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \end{array}\right] \end{array},$$

where column operations from 1:2 to 2:3 and to 2:4 are impermissible.

4. Now our target column is 2:4. In a similar case as discussed in the case of $CL_3(bf)$, indecomposable neighboring two identity blocks will appear on the process when $v_* = (2{:}2_1, 2{:}4_2)$. Although $(2{:}2_1, 2{:}4_2)$ is chosen as the target v_* twice during the procedures on this column, the issue happens at the first time. The block is of the form $\begin{matrix} \\ 2{:}2_1 \end{matrix}\begin{matrix} 2{:}4_2 & 1{:}2_1 \\ \end{matrix}\begin{bmatrix} E & E \end{bmatrix}$, expressed by the dimension vector $\begin{smallmatrix} 0100 \\ 1211 \end{smallmatrix}$.

   Furthermore in this column, one observes indecomposable neighboring three identity blocks $\begin{matrix} \\ 2{:}2_1 \\ 4{:}4_2 \end{matrix}\begin{matrix} 2{:}4_2 & 1{:}2_1 \\ \end{matrix}\begin{bmatrix} E & E \\ E & \emptyset \end{bmatrix}$ at the second time of $v_* = (2{:}2_1, 2{:}4_2)$. This is isomorphic to copies of the indecomposable persistence module

$$\begin{array}{ccccccc} 0 & \longleftarrow & K & \longrightarrow & 0 & \longleftarrow & K \\ \uparrow & & \uparrow{\scriptstyle [1\ 1]} & & \uparrow & & \uparrow{\scriptstyle \mathrm{id}} \\ K & \xleftarrow{[1\ 0]} & K^2 & \xrightarrow{[0\ 1]} & K & \xleftarrow{\mathrm{id}} & K \end{array},$$

   and expressed by the dimension vector $\begin{smallmatrix} 0101 \\ 1211 \end{smallmatrix}$.

   After all the procedures on this column, we obtain $\begin{smallmatrix} 0111 \\ 0111 \end{smallmatrix}$, $\begin{smallmatrix} 0100 \\ 0111 \end{smallmatrix}$, $\begin{smallmatrix} 0100 \\ 1211 \end{smallmatrix}$, $\begin{smallmatrix} 0000 \\ 0111 \end{smallmatrix}$, $\begin{smallmatrix} 0101 \\ 1211 \end{smallmatrix}$, and $\begin{smallmatrix} 0001 \\ 0111 \end{smallmatrix}$. The block matrix form is now

$$\begin{array}{c} \\ 2{:}2_1 \\ 2{:}2_2 \\ 2{:}2_3 \\ 4{:}4_1 \\ 4{:}4_2 \\ 1{:}2 \\ 2{:}4 \\ 1{:}4 \\ 2{:}3 \\ 1{:}3 \\ 3{:}4 \\ 1{:}1 \\ 3{:}3 \end{array}\begin{array}{c} \begin{array}{cccccccc} 3{:}3 & 1{:}1 & 3{:}4 & 1{:}3 & 2{:}3 & 1{:}4 & 2{:}4 & 1{:}2 \end{array} \\ \begin{bmatrix} \emptyset & \emptyset & \emptyset & 0 & * & 0 & 0 & E \\ \emptyset & \emptyset & \emptyset & 0 & 0 & * & E & 0 \\ \emptyset & \emptyset & \emptyset & * & * & * & 0 & 0 \\ \emptyset & \emptyset & 0 & \emptyset & \emptyset & 0 & E & \emptyset \\ \emptyset & \emptyset & * & \emptyset & \emptyset & * & 0 & \emptyset \\ \emptyset & * & \emptyset & * & \emptyset & * & \emptyset & 0 \\ * & \emptyset & * & * & * & * & 0 & \emptyset \\ * & * & * & * & \emptyset & * & \emptyset & \emptyset \\ * & \emptyset & \emptyset & * & * & \emptyset & \emptyset & \emptyset \\ * & * & \emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset \\ * & \emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \\ \emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \\ * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \end{bmatrix} \end{array}.$$

5. Next, procedures on column 1:4 yield $\begin{smallmatrix} 1111 \\ 1111 \end{smallmatrix}$, $\begin{smallmatrix} 0101 \\ 1222 \end{smallmatrix}$, $\begin{smallmatrix} 0101 \\ 0111 \end{smallmatrix}$, $\begin{smallmatrix} 0000 \\ 1111 \end{smallmatrix}$, $\begin{smallmatrix} 0001 \\ 1111 \end{smallmatrix}$, $\begin{smallmatrix} 1101 \\ 1111 \end{smallmatrix}$, and $\begin{smallmatrix} 1100 \\ 1111 \end{smallmatrix}$. The

matrix is of the form:

$$\begin{array}{c} \\ 2{:}2_1 \\ 2{:}2_2 \\ 2{:}2_3 \\ 2{:}2_4 \\ 4{:}4_1 \\ 4{:}4_2 \\ 1{:}2_1 \\ 1{:}2_2 \\ 2{:}4_1 \\ 2{:}4_2 \\ 2{:}4_3 \\ 1{:}4 \\ 2{:}3 \\ 1{:}3 \\ 3{:}4 \\ 1{:}1 \\ 3{:}3 \end{array} \begin{array}{c} \begin{array}{cccccccccc} 3{:}3 & 1{:}1 & 3{:}4 & 1{:}3 & 2{:}3 & 1{:}4_1 & 1{:}4_2 & 1{:}4_3 & 1{:}4_4 & 1{:}2 \end{array} \\ \begin{bmatrix} \emptyset & \emptyset & \emptyset & 0 & * & 0 & 0 & 0 & 0 & E \\ \emptyset & \emptyset & \emptyset & 0 & * & 0 & 0 & 0 & E & 0 \\ \emptyset & \emptyset & \emptyset & 0 & * & 0 & 0 & E & 0 & 0 \\ \emptyset & \emptyset & \emptyset & * & * & 0 & 0 & 0 & 0 & 0 \\ \emptyset & \emptyset & 0 & \emptyset & \emptyset & 0 & 0 & E & 0 & \emptyset \\ \emptyset & \emptyset & * & \emptyset & \emptyset & 0 & 0 & 0 & 0 & \emptyset \\ \emptyset & 0 & \emptyset & * & \emptyset & E & 0 & 0 & 0 & 0 \\ \emptyset & * & \emptyset & * & \emptyset & 0 & 0 & 0 & 0 & 0 \\ 0 & \emptyset & 0 & 0 & * & E & 0 & 0 & 0 & \emptyset \\ 0 & \emptyset & 0 & 0 & * & 0 & E & 0 & 0 & \emptyset \\ * & \emptyset & * & * & * & 0 & 0 & 0 & 0 & \emptyset \\ * & * & * & * & \emptyset & 0 & 0 & 0 & 0 & \emptyset \\ * & \emptyset & \emptyset & * & * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \\ * & * & \emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \\ * & \emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \\ \emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \\ * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \end{bmatrix} \end{array}.$$

6. We next turn to column 2:3, and finally encounter an erasable identity block. This occurs immediately after transforming $(2{:}4_1, 2{:}3_2)$ to Smith normal form. Before this, some other blocks in column 3:3 are processed without issue. Below, we present the relevant block of the matrix before and after transforming v_* to Smith normal form, and the block partition is updated to split the identity blocks:

$$\begin{array}{c} \\ 1{:}2_1 \\ 2{:}4_1 \\ 2{:}4_2 \end{array} \begin{array}{c} \begin{array}{ccc} 2{:}3_2 & 1{:}4_1 & 1{:}4_2 \end{array} \\ \begin{bmatrix} \emptyset & E & 0 \\ v_* & E & 0 \\ E & 0 & E \end{bmatrix} \end{array} \cong \begin{array}{c} \\ 1{:}2_1 \\ 1{:}2_2 \\ 2{:}4_1 \\ 2{:}4_2 \\ 2{:}4_3 \\ 2{:}4_4 \end{array} \begin{array}{c} \begin{array}{cccccc} 2{:}3_2 & 2{:}3_3 & 1{:}4_1 & 1{:}4_2 & 1{:}4_3 & 1{:}4_4 \end{array} \\ \begin{bmatrix} \emptyset & \emptyset & E & 0 & 0 & 0 \\ \emptyset & \emptyset & 0 & E & 0 & 0 \\ E & 0 & E & 0 & 0 & 0 \\ 0 & 0 & 0 & E & 0 & 0 \\ E & 0 & 0 & 0 & E & 0 \\ 0 & E & 0 & 0 & 0 & E \end{bmatrix} \end{array}.$$

The identity block $(2{:}4_3, 2{:}3_2)$ is erasable by

a) additions from the row $2{:}4_1$ to the row $2{:}4_3$, and

b) additions from the column $1{:}4_3$ to the column $1{:}4_1$ to zero out the side effect at $(2{:}4_3, 1{:}4_1)$.

Here $\begin{smallmatrix} 0110 \\ 0110 \end{smallmatrix}$, $\begin{smallmatrix} 0111 \\ 1111 \end{smallmatrix}$, $\begin{smallmatrix} 1211 \\ 1221 \end{smallmatrix}$, $\begin{smallmatrix} 0100 \\ 0110 \end{smallmatrix}$, $\begin{smallmatrix} 0101 \\ 1221 \end{smallmatrix}$, $\begin{smallmatrix} 0101 \\ 1111 \end{smallmatrix}$, $\begin{smallmatrix} 0100 \\ 1221 \end{smallmatrix}$, $\begin{smallmatrix} 0100 \\ 1111 \end{smallmatrix}$, $\begin{smallmatrix} 0100 \\ 1210 \end{smallmatrix}$, $\begin{smallmatrix} 0100 \\ 1100 \end{smallmatrix}$, $\begin{smallmatrix} 0000 \\ 0110 \end{smallmatrix}$, $\binom{1211}{1221}$, $\binom{0111}{1111}$, $\begin{smallmatrix} 0211 \\ 1221 \end{smallmatrix}$, and $\begin{smallmatrix} 0111 \\ 1221 \end{smallmatrix}$ are extracted. We remark that the dimension vectors $\begin{smallmatrix} 1221 \\ 0121 \end{smallmatrix}$ and $\begin{smallmatrix} 0110 \\ 0111 \end{smallmatrix}$ are extracted twice.

$$
\begin{array}{c}
\\
2{:}2\\
4{:}4\\
1{:}2_1\\
1{:}2_2\\
2{:}4_1\\
2{:}4_2\\
2{:}4_3\\
1{:}4\\
2{:}3\\
1{:}3\\
3{:}4\\
1{:}1\\
3{:}3
\end{array}
\begin{array}{c}
\begin{array}{cccccc} 3{:}3 & 1{:}1 & 3{:}4 & 1{:}3 & 2{:}3 & 1{:}4 \end{array}\\
\left[\begin{array}{cccccc}
\emptyset & \emptyset & \emptyset & * & 0 & 0\\
\emptyset & \emptyset & * & \emptyset & \emptyset & 0\\
\emptyset & 0 & \emptyset & * & \emptyset & E\\
\emptyset & * & \emptyset & * & \emptyset & 0\\
0 & \emptyset & 0 & 0 & 0 & E\\
0 & \emptyset & * & 0 & E & 0\\
* & \emptyset & * & * & 0 & 0\\
* & * & * & * & \emptyset & 0\\
* & \emptyset & \emptyset & * & 0 & \emptyset\\
* & * & \emptyset & * & \emptyset & \emptyset\\
* & \emptyset & * & \emptyset & \emptyset & \emptyset\\
\emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset\\
* & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset
\end{array}\right]
\end{array}.
$$

7. The work afterward is routine. By procedures on column 1:3, $\begin{smallmatrix}1110\\1110\end{smallmatrix}$, $\begin{smallmatrix}1100\\1110\end{smallmatrix}$, $\begin{smallmatrix}1211\\2221\end{smallmatrix}$, $\begin{smallmatrix}1211\\1111\end{smallmatrix}$, $\begin{smallmatrix}0100\\1110\end{smallmatrix}$, $\begin{smallmatrix}0000\\1110\end{smallmatrix}$, $\begin{smallmatrix}0100\\0000\end{smallmatrix}$, $\begin{smallmatrix}1210\\1110\end{smallmatrix}$, and $\begin{smallmatrix}0110\\1110\end{smallmatrix}$ are extracted. We are left with

$$
\begin{array}{c}
\\
4{:}4\\
1{:}2_1\\
1{:}2_2\\
2{:}4_1\\
2{:}4_2\\
2{:}4_3\\
2{:}4_4\\
1{:}4_1\\
1{:}4_2\\
1{:}4_3\\
2{:}3_1\\
2{:}3_2\\
1{:}3\\
3{:}4\\
1{:}1\\
3{:}3
\end{array}
\begin{array}{c}
\begin{array}{cccccccc} 3{:}3 & 1{:}1 & 3{:}4 & 1{:}3_1 & 1{:}3_2 & 1{:}3_3 & 1{:}3_4 & 2{:}3 \end{array}\\
\left[\begin{array}{cccccccc}
\emptyset & \emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset\\
\emptyset & 0 & \emptyset & 0 & 0 & E & 0 & \emptyset\\
\emptyset & * & \emptyset & 0 & 0 & 0 & 0 & \emptyset\\
0 & \emptyset & * & 0 & 0 & 0 & 0 & E\\
0 & \emptyset & * & 0 & 0 & E & 0 & 0\\
0 & \emptyset & * & 0 & 0 & 0 & E & 0\\
* & \emptyset & * & 0 & 0 & 0 & 0 & 0\\
0 & 0 & * & 0 & E & 0 & 0 & \emptyset\\
* & 0 & * & E & 0 & 0 & 0 & \emptyset\\
* & * & * & 0 & 0 & 0 & 0 & \emptyset\\
0 & \emptyset & \emptyset & E & 0 & 0 & 0 & 0\\
* & \emptyset & \emptyset & 0 & 0 & 0 & 0 & 0\\
* & * & \emptyset & 0 & 0 & 0 & 0 & \emptyset\\
* & \emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset\\
\emptyset & * & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset\\
* & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset & \emptyset
\end{array}\right]
\end{array}.
$$

8. Next, procedures on column 3:4 yield $\begin{smallmatrix}0011\\0011\end{smallmatrix}$, $\begin{smallmatrix}1221\\1121\end{smallmatrix}$, $\begin{smallmatrix}1111\\1110\end{smallmatrix}$, $\begin{smallmatrix}0111\\0011\end{smallmatrix}$, $\begin{smallmatrix}0111\\1110\end{smallmatrix}$, $\begin{smallmatrix}1211\\1121\end{smallmatrix}$, $\begin{smallmatrix}0111\\0121\end{smallmatrix}$, $\begin{smallmatrix}0111\\0110\end{smallmatrix}$, $\begin{smallmatrix}0001\\0011\end{smallmatrix}$, $\begin{smallmatrix}0000\\0011\end{smallmatrix}$,

$\begin{smallmatrix}0001\\0000\end{smallmatrix}$, $\binom{1211}{1121}$, $\begin{smallmatrix}1211\\1110\end{smallmatrix}$, $\binom{0111}{1110}$, $\begin{smallmatrix}0111\\1121\end{smallmatrix}$, $\begin{smallmatrix}1222\\1121\end{smallmatrix}$, and $\begin{smallmatrix}1111\\1121\end{smallmatrix}$, where $\begin{smallmatrix}1211\\1121\end{smallmatrix}$ and $\begin{smallmatrix}0111\\1110\end{smallmatrix}$ are extracted twice.

$$\begin{array}{c} \\ 1{:}2 \\ 2{:}4 \\ 1{:}4_1 \\ 1{:}4_2 \\ 1{:}4_3 \\ 2{:}3_1 \\ 2{:}3_2 \\ 1{:}3 \\ 3{:}4 \\ 1{:}1 \\ 3{:}3 \end{array} \begin{array}{c} \begin{array}{cccc} 3{:}3 & 1{:}1 & 3{:}4 & 1{:}3 \end{array} \\ \left[\begin{array}{cccc} \emptyset & * & \emptyset & 0 \\ * & \emptyset & 0 & 0 \\ * & 0 & 0 & E \\ 0 & * & E & 0 \\ * & * & 0 & 0 \\ 0 & \emptyset & \emptyset & E \\ * & \emptyset & \emptyset & 0 \\ * & * & \emptyset & 0 \\ * & \emptyset & 0 & \emptyset \\ \emptyset & * & \emptyset & \emptyset \\ * & \emptyset & \emptyset & \emptyset \end{array}\right] \end{array}$$

9. Subsequently, procedures on column 1:1 extract $\begin{smallmatrix}1000\\1000\end{smallmatrix}$, $\begin{smallmatrix}1000\\0000\end{smallmatrix}$, $\begin{smallmatrix}1111\\1011\end{smallmatrix}$, $\begin{smallmatrix}1111\\0011\end{smallmatrix}$, $\begin{smallmatrix}1100\\1000\end{smallmatrix}$, $\begin{smallmatrix}0000\\1000\end{smallmatrix}$, and $\begin{smallmatrix}1100\\0000\end{smallmatrix}$. The block matrix is now

$$\begin{array}{c} \\ 2{:}4 \\ 1{:}4_1 \\ 1{:}4_2 \\ 1{:}4_3 \\ 2{:}3_1 \\ 2{:}3_2 \\ 1{:}3_1 \\ 1{:}3_2 \\ 3{:}4 \\ 3{:}3 \end{array} \begin{array}{c} \begin{array}{cccc} 3{:}3 & 1{:}1_1 & 1{:}1_2 & 1{:}3 \end{array} \\ \left[\begin{array}{cccc} * & \emptyset & \emptyset & 0 \\ * & 0 & 0 & E \\ * & 0 & E & 0 \\ * & 0 & 0 & 0 \\ 0 & \emptyset & \emptyset & E \\ * & \emptyset & \emptyset & 0 \\ * & E & 0 & 0 \\ * & 0 & 0 & 0 \\ * & \emptyset & \emptyset & \emptyset \\ * & \emptyset & \emptyset & \emptyset \end{array}\right] \end{array}.$$

10. Finally we are on the last column 4:4. Here $\begin{smallmatrix}0010\\0010\end{smallmatrix}$, $\begin{smallmatrix}0010\\0000\end{smallmatrix}$, $\begin{smallmatrix}0011\\0000\end{smallmatrix}$, $\begin{smallmatrix}1110\\0010\end{smallmatrix}$, $\begin{smallmatrix}1111\\0010\end{smallmatrix}$, $\begin{smallmatrix}1111\\1010\end{smallmatrix}$, $\begin{smallmatrix}1221\\1120\end{smallmatrix}$, $\begin{smallmatrix}1221\\1110\end{smallmatrix}$, $\begin{smallmatrix}0111\\0010\end{smallmatrix}$, $\begin{smallmatrix}0000\\0010\end{smallmatrix}$, $\begin{smallmatrix}0111\\0000\end{smallmatrix}$, $\begin{smallmatrix}1221\\0010\end{smallmatrix}$, $\begin{smallmatrix}1221\\1010\end{smallmatrix}$, $\begin{smallmatrix}1111\\1000\end{smallmatrix}$, $\begin{smallmatrix}0110\\0010\end{smallmatrix}$, $\begin{smallmatrix}2221\\1010\end{smallmatrix}$, $\begin{smallmatrix}1110\\1010\end{smallmatrix}$, $\begin{smallmatrix}1111\\0000\end{smallmatrix}$, $\begin{smallmatrix}1121\\0010\end{smallmatrix}$, $\begin{smallmatrix}1110\\0000\end{smallmatrix}$, $\begin{smallmatrix}1121\\1010\end{smallmatrix}$, $\begin{smallmatrix}1110\\1000\end{smallmatrix}$, $\begin{smallmatrix}0121\\0010\end{smallmatrix}$, $\begin{smallmatrix}0110\\0000\end{smallmatrix}$, and $\begin{smallmatrix}0011\\0010\end{smallmatrix}$ are extracted.

3.3. Discussion

An example can be given of how the proof breaks down with the infinite type ($n \geq 5$), in particular the case $CL_5(ffff)$. We only need to consider the subproblem of the block matrix problem spanned by the rows $1{:}4, 2{:}3$ and columns $2{:}5, 3{:}4$:

$$\begin{array}{c} \\ 1{:}4 \\ 2{:}3 \end{array} \begin{array}{c} \begin{array}{cc} 3{:}4 & 2{:}5 \end{array} \\ \left[\begin{array}{cc} * & * \\ * & * \end{array}\right] \end{array}.$$

Note that operations between rows 1:4 and 2:3 are impermissible. Similarly, operations between columns 2:5 and 3:4 are impermissible. After applying the procedure, we eventually obtain

$$\begin{matrix} & \scriptstyle 3{:}4 & \scriptstyle 2{:}5 \\ \scriptstyle 1{:}4 \\ \scriptstyle 2{:}3 \end{matrix}\begin{bmatrix} A & E \\ E & E \end{bmatrix}$$

as a subproblem, where A is an unerasable unprocessed submatrix and E are processed identity matrices of the same size.

The procedure breaks down when attempting to transform the unprocessed block A to Smith normal form. Starting with row operations on 1:4, we obtain side effects on the neighboring block. By fixing those, we obtain side effects on the block below and others as

$$\begin{matrix} & \scriptstyle 3{:}4 & \scriptstyle 2{:}5 \\ \scriptstyle 1{:}4 \\ \scriptstyle 2{:}3 \end{matrix}\begin{bmatrix} PA & P \\ E & E \end{bmatrix} \cong \begin{matrix} & \scriptstyle 3{:}4 & \scriptstyle 2{:}5 \\ \scriptstyle 1{:}4 \\ \scriptstyle 2{:}3 \end{matrix}\begin{bmatrix} PA & E \\ E & P^{-1} \end{bmatrix} \cong \begin{matrix} & \scriptstyle 3{:}4 & \scriptstyle 2{:}5 \\ \scriptstyle 1{:}4 \\ \scriptstyle 2{:}3 \end{matrix}\begin{bmatrix} PA & E \\ P & E \end{bmatrix} \cong \begin{matrix} & \scriptstyle 3{:}4 & \scriptstyle 2{:}5 \\ \scriptstyle 1{:}4 \\ \scriptstyle 2{:}3 \end{matrix}\begin{bmatrix} PAP^{-1} & E \\ E & E \end{bmatrix}.$$

Next, we comment on existing methods and the computational aspects of our work. Here, we take the persistence module as input and measure the computational cost with respect to its size. Requiring the explicit computation of the persistence module may be wasteful. Indeed, classical algorithms for persistence work directly on the level of diagrams of simplicial complexes (for example [28, 6] for filtrations and zigzags, respectively). It is of interest to find similar algorithms that compute an indecomposable decomposition of the persistence module directly, without explicit computation of the persistence module.

The problem of computing an indecomposable decomposition of a module M over a finite dimensional K-algebra A has been well-studied [9, 24]. For example, assuming that K is a finite field, [9] demonstrates that a polynomial time algorithm exists that is capable of finding an indecomposable decomposition of M. First, one computes the endomorphism algebra $\mathrm{End}_A(M) = \mathrm{Hom}(M, M)$. Then, a complete set of primitive orthogonal idempotents $\pi_1, \ldots, \pi_\ell$ is computed. The submodules $\pi_i(M)$, for $i = 1, \ldots, \ell$, provide a decomposition of M into indecomposables.

One straightforward means of computing $\mathrm{End}_A(M)$ is by solving a system of linear equations. Let $M = (M_a, \varphi_\alpha)_{a \in Q_0, \alpha \in Q_1}$ be a representation of a bound quiver (Q, P), $d_a = \dim M_a$, and $\hat{d} = \max_{a \in Q_0} d_a$. We assume that we have fixed bases for M_a, and that the actions φ_α are given in terms of K-matrices. For each vertex $a \in Q_0$, we set up a $d_a \times d_a$ K-matrix of unknowns X_a, and solve the condition given in Eq. (2.1) that $X_b \varphi_\alpha - \varphi_\alpha X_a = 0$ for each arrow $\alpha : a \to b$, giving a linear system of $\sum_{a \in Q_0} d_a^2 = \mathcal{O}(\hat{d}^2)$ unknowns and $\sum_{(\alpha : a \to b) \in Q_1} d_a d_b = \mathcal{O}(\hat{d}^2)$ equations. Consequently, it seems to require $\mathcal{O}(\hat{d}^6)$ computational time, even before the rest of the computations (finding primitive orthogonal idempotents and computing images). Additional clever methods of computing the endomorphism rings exist, for example by reducing the number of equations and unknowns by first computing a generating system for the module [25] or using Gröbner basis methods [20].

Finally, let us provide an estimate of the computational complexity of our procedure. First, we recapitulate the matrix formalism in Subsection 3.1.2. Given a persistence module M over a commutative ladder with $n < 5$, writing it as an arrow $F(M) : V \to W$ is straightforward. Then, we decompose its upper and lower rows while keeping track of basis changes to get the forms of the matrices of arrow Φ relative to the new basis, costing $\mathcal{O}(\hat{d}^3)$.

Let us analyze the size of the corresponding block matrix problem of M. Recall that the blocks are of size $m'_{c,d} \times m_{a,b}$ for $\mathbb{I}[a,b] \trianglerighteq \mathbb{I}[c,d]$, where the numbers $m_{a,b}$ and $m'_{c,d}$ are determined by the decomposition of $F(M)$ as in Eq. (3.3). It is clear that $m_{a,b}$ and $m'_{c,d}$ are bounded above by $\hat{d}$. If that were not true, then there would be a vertex x of the commutative ladder with $d_x > \hat{d}$, which is a contradiction. We also compute:

$$n\hat{d} \geq \dim V = \sum_{1 \leq a \leq b \leq n} \dim \mathbb{I}[a,b]^{m_{a,b}} = \sum_{1 \leq a \leq b \leq n} m_{a,b}(b-a+1) \geq \sum_{1 \leq a \leq b \leq n} m_{a,b}$$

from Eq. (3.2), and a similar estimate for the sum of the $m'_{c,d}$. Thus, the total size of the block matrix problem is bounded above by $n\hat{d}$.

The above analysis considers the size of the block matrix problem by counting actual rows and columns of K-scalars. In contrast, the number of blocks b themselves is not dependent on the dimension $\hat{d}$, but rather depends only on the orientation τ_n and what operations have been performed so far.

Next, we look at Algorithm 1. Each query for side effects in Algorithm 2 costs $\mathcal{O}(b)$. By the proof of Theorem 3.2.3, Lines 7 and 8 of Algorithm 1 do not lead to infinite recursion, thus only a finite number of such queries and matrix operations are needed for Algorithm 3. This number depends not on $\hat{d}$ but only on the current arrangement of block statuses of the block matrix problem.

Furthermore, the check for erasability and construction of a process tree p by ERASABLE in Line 10 of Algorithm 1 is independent of $\hat{d}$. Indeed, the functions ROW_ERASABLE and COL_ERASABLE do not perform any matrix operations. Rather, they simply query the arrangement of matrix statuses and permissible operations in the block matrix problem.

The actual operations on the block matrix problem are given by: transforming a block to Smith normal form, fixing one side effect, zeroing out v_t by v_E for one vertex (v_t, v_E) in the process tree p. Each of these costs $\mathcal{O}(n\hat{d}^3)$. Thus, the total cost is simply the number of operations C needed, which is independent of $\hat{d}$, times $\mathcal{O}(n\hat{d}^3)$, yielding an overall estimate of $\mathcal{O}(Cn\hat{d}^3)$.

Chapter 4.

The Persistent Homology of a Sampled Map

This chapter is broadly divided into two parts. The former part as Section 4.1 presents a new definition of the homology induced map of a correspondence. This definition is formalized by decomposing representations of the quiver $A_3(bf)$, and gives us the first step in the latter part. At the latter part from Section 4.2 on, we discuss how to construct a persistence module from a sampled map and decompose it. Techniques described there are stimulated by block matrix form discussed in Chapter 3, and enable topological data analysis on time series data. Ahead of the thesis, the content of this chapter was uploaded as a preprint on arXiv:

[27] H. Takeuchi, The Persistent Homology of a Sampled Map: From a Viewpoint of Quiver Representations, arXiv:1810.11774 (2018).

4.1. The Induced Maps via Quiver Representations

Recall that a correspondence $F \subset X \times Y$ induces the diagram $HX \xleftarrow{p_*} HF \xrightarrow{q_*} HY$ via the canonical projections $p\colon F \to X$ and $q\colon F \to Y$. As a representation of the $A_3(bf)$ type quiver, this diagram can be decomposed into a direct sum of interval representations:

$$(HX \xleftarrow{p_*} HF \xrightarrow{q_*} HY) \cong \bigoplus_{1 \le b \le d \le 3} \mathbb{I}[b,d]^{m_{b,d}}.$$

This indecomposable decomposition can be written as the diagram

$$\bigoplus_{1 \le b \le d \le 3} \mathbb{I}[b,d]^{m_{b,d}} = \begin{array}{ccccc} HX & \xleftarrow{p_*} & HF & \xrightarrow{q_*} & HY \\ {\scriptstyle h_X}\downarrow{\scriptstyle \cong} & & \downarrow{\scriptstyle \cong} & & {\scriptstyle h_Y}\downarrow{\scriptstyle \cong} \\ K^{\dim HX} & \longleftarrow & K^{\dim HF} & \xrightarrow{q^K} & K^{\dim HY}. \end{array}$$

The choice of bases gives us a relationship between bases of HX and HY, which can be regarded as a map from HX to HY. For example, an interval representation $\mathbb{I}[1,2]$ assigns an element of the standard basis of $K^{\dim HX}$ to 0 in $K^{\dim HY}$. Therefore, non-trivial assignment happens only on the interval representations $\mathbb{I}[1,3] = (K \xleftarrow{\mathrm{id}_K} K \xrightarrow{\mathrm{id}_K} K)$.

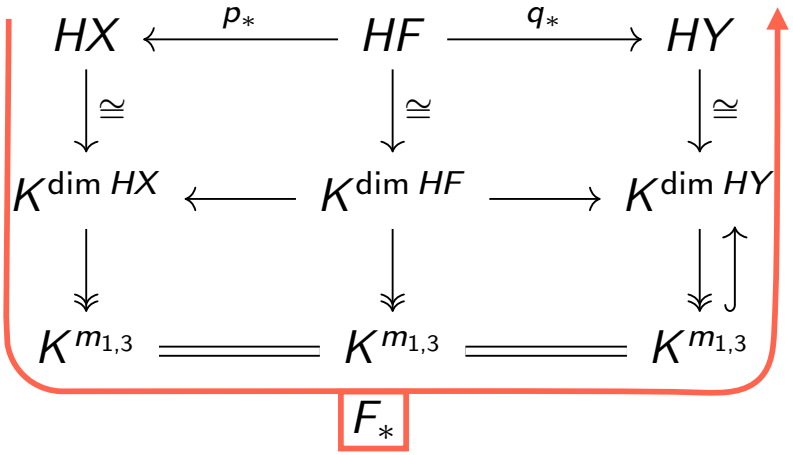


Figure 4.1.: An overview of our definition of the induced map F_* of a correspondence. The isomorphism from the first row to the second row is an indecomposable decomposition. The inclusion map on the right-hand side is the canonical injection of the vector space.

By regarding the other interval representations as 0 maps from HX to HY, we can define a map $\iota_Y \circ \pi_X \colon K^{\dim HX} \to K^{\dim HY}$ factoring the interval $\mathbb{I}[1,3]$

$$\begin{array}{ccc} \bigoplus_{1\le b\le d\le 3} \mathbb{I}[b,d]^{m_{b,d}} = & K^{\dim HX} \longleftarrow K^{\dim HF} \xrightarrow{q^K} & K^{\dim HY} \\ \downarrow & \downarrow \pi_X \quad \downarrow \pi_F \quad \downarrow \iota_Y & \\ \mathbb{I}[1,3]^{m_{1,3}} = & K^{m_{1,3}} = K^{m_{1,3}} = & K^{m_{1,3}}, \end{array}$$

where the arrows $\twoheadrightarrow$ are the canonical projections of the vector spaces, and the morphism ι_Y is the canonical injection of the vector space. Composing the path of morphisms, we can redefine the induced map of F as

$$F_* := h_Y^{-1} \circ \iota_Y \circ \pi_X \circ h_X \colon HX \to HY.$$

This definition does not require the two assumptions mentioned in Definition 1.2.3, and when the two assumptions are satisfied, our definition coincides with the original definition $q_* \circ p_*^{-1}$ by the following theorem.

Theorem 4.1.1. Define $F_* := h_Y^{-1} \circ \iota_Y \circ \pi_X \circ h_X$. If F is homologically complete and homologically consistent, then $F_* = q_* \circ p_*^{-1}$.

Proof. The claim to prove is

$$q_* \circ p_*^{-1} = h_Y^{-1} \circ \iota_Y \circ \pi_X \circ h_X.$$

Since the morphism p_* is surjective, this is equivalent to

$$q_* \circ p_*^{-1} \circ p_* = h_Y^{-1} \circ \iota_Y \circ \pi_X \circ h_X \circ p_*.$$

In addition, $q_* = q_* \circ p_*^{-1} \circ p_*$ because of the homological consistency $q_*(\mathrm{Ker}(p_*)) = 0$ Therefore, what must be proven here is

$$q_* = h_Y^{-1} \circ \iota_Y \circ \pi_X \circ h_X \circ p_*.$$

By chasing the diagram of Figure 4.1, this equation results in

$$q^K = \iota_Y \circ \pi_F.$$

The standard basis of $K^{\dim HF}$ corresponds to the standard bases of the four intervals $\mathbb{I}[2,3]$, $\mathbb{I}[2,2]$, $\mathbb{I}[1,2]$, and $\mathbb{I}[1,3]$. Here we remark that the homological consistency $q_*(\mathrm{Ker}(p_*)) = 0$ is equivalent to $m_{2,3} = 0$, namely $\mathbb{I}[2,3]$ does not exist as a direct summand. Moreover, the basis corresponding to $\mathbb{I}[2,2]$ and $\mathbb{I}[1,2]$ is mapped to 0 by both q^K and $\iota_Y \circ \pi_F$. By definition it is clear that $q^K(a) = \iota_Y \circ \pi_F(a)$ holds for each element a of the standard basis of $K^{\dim HF}$ corresponding to the standard bases of $\mathbb{I}[1,3]$. □

Theorems 1.2.4 and 1.2.5 hold also under this definition. We can prove them more concisely by focusing on indecomposable representations of the commutative ladder $CL(bf)$. Since Theorem 1.2.5 is a generalized statement of Theorem 1.2.4, we prove only Theorem 1.2.5. Let us reproduce here Theorem 1.2.5.

Theorem 1.2.5 ([21, Theorem 4.6])**.** If a correspondence F is homologically consistent, a correspondence G is homologically complete, and $G \subset F$, then F_* and G_* are well-defined and $G_* = F_*$.

Proof. Let $p_*^F \colon HF \to HX$, $q_*^F \colon HF \to HY$, $p_*^G \colon HG \to HX$, and $q_*^G \colon HG \to HY$ be appropriate morphisms induced by canonical projections of $X \times Y$. The assumptions induce the following commutative diagram for a representation of the commutative ladder $CL(bf)$.

$$\begin{array}{ccccc} HX & \xleftarrow{p_*^F} & HF & \xrightarrow{q_*^F} & HY \\ \| & & \uparrow & & \| \\ HX & \xleftarrow{p_*^G} & HG & \xrightarrow{q_*^G} & HY \end{array} \tag{4.1}$$

where the center morphism $HG \to HF$ is induced by the inclusion $G \subset F$.

Let the diagram

$$\begin{array}{ccccc} M_1' & \xleftarrow{p_2} & M_2' & \xrightarrow{q_2} & M_3' \\ {\scriptstyle r_1}\uparrow & & {\scriptstyle r_2}\uparrow & & {\scriptstyle r_3}\uparrow \\ M_1 & \xleftarrow{p_1} & M_2 & \xrightarrow{q_1} & M_3 \end{array}$$

be a direct summand of the representation (4.1). By the definition of indecomposable decomposition, the properties of the morphisms of the total representation are inherited by the morphisms in its direct summand. First, the morphisms r_1 and r_3 should be isomorphic. Second, the homological consistency $q_*^F(\mathrm{Ker}\, p_*^F) = 0$ inherits $q_2(\mathrm{Ker}\, p_2) = 0$, which means that indecomposable decomposition of the upper representation

$$M_1' \xleftarrow{p_2} M_2' \xrightarrow{q_2} M_3'$$

does not yield $\mathbb{I}[2,3]$. Similarly, the homological completeness of G implies that the interval $\mathbb{I}[1,1]$ does not appear in the lower side.

With these three properties in mind, let us look at the Auslander–Reiten quiver of the commutative ladder $CL(bf)$ (Figure 2.1). Here we use it just as a list of indecomposable representations of $CL(bf)$. The three properties reduce the number of candidates of indecomposables from 30 to 6. First, since r_1 and r_3 are isomorphisms, the dimensions of the left (right) upper and left (right, respectively) lower vectors should be the same. Therefore, from left to right in Figure 2.1, the indecomposables $\begin{smallmatrix}100\\000\end{smallmatrix}$, $\begin{smallmatrix}001\\000\end{smallmatrix}$, $\begin{smallmatrix}111\\000\end{smallmatrix}$, $\begin{smallmatrix}111\\100\end{smallmatrix}$, $\begin{smallmatrix}111\\001\end{smallmatrix}$, $\begin{smallmatrix}011\\000\end{smallmatrix}$, $\begin{smallmatrix}110\\000\end{smallmatrix}$, $\begin{smallmatrix}000\\001\end{smallmatrix}$, $\begin{smallmatrix}000\\100\end{smallmatrix}$, $\begin{smallmatrix}110\\111\end{smallmatrix}$, $\begin{smallmatrix}011\\111\end{smallmatrix}$, $\begin{smallmatrix}010\\111\end{smallmatrix}$, $\begin{smallmatrix}010\\110\end{smallmatrix}$, $\begin{smallmatrix}000\\111\end{smallmatrix}$, $\begin{smallmatrix}010\\011\end{smallmatrix}$, $\begin{smallmatrix}010\\121\end{smallmatrix}$, $\begin{smallmatrix}000\\011\end{smallmatrix}$, and $\begin{smallmatrix}000\\110\end{smallmatrix}$ cannot appear. Second, since the interval $\mathbb{I}[2,3]$ should not appear in the upper side, $\begin{smallmatrix}011\\001\end{smallmatrix}$ and $\begin{smallmatrix}011\\011\end{smallmatrix}$ cannot appear. Moreover, as portrayed in Diagram (2.4), the upper side of $\begin{smallmatrix}121\\111\end{smallmatrix}$ is the direct sum of $\mathbb{I}[1,2]$ and $\mathbb{I}[2,3]$, hence this cannot appear either. Finally, since the interval $\mathbb{I}[1,1]$ should not appear in the lower side, $\begin{smallmatrix}100\\100\end{smallmatrix}$, $\begin{smallmatrix}111\\101\end{smallmatrix}$ and $\begin{smallmatrix}110\\100\end{smallmatrix}$ cannot appear.

Eventually, the indecomposables of Diagram (4.1) can only be the other remaining six types $\begin{smallmatrix}001\\001\end{smallmatrix}$, $\begin{smallmatrix}111\\111\end{smallmatrix}$, $\begin{smallmatrix}010\\000\end{smallmatrix}$, $\begin{smallmatrix}110\\110\end{smallmatrix}$, $\begin{smallmatrix}010\\010\end{smallmatrix}$, and $\begin{smallmatrix}000\\010\end{smallmatrix}$, that is,

$$\begin{array}{ccccc} HX & \xleftarrow{p_*^F} & HF & \xrightarrow{q_*^F} & HY \\ \| & & \uparrow & & \| \\ HX & \xleftarrow{p_*^G} & HG & \xrightarrow{q_*^G} & HY \end{array} \cong \begin{pmatrix}001\\001\end{pmatrix}^{m_1} \oplus \begin{pmatrix}111\\111\end{pmatrix}^{m_2} \oplus \begin{pmatrix}010\\000\end{pmatrix}^{m_3} \oplus \begin{pmatrix}110\\110\end{pmatrix}^{m_4} \oplus \begin{pmatrix}010\\010\end{pmatrix}^{m_5} \oplus \begin{pmatrix}000\\010\end{pmatrix}^{m_6}$$

for some $m_i \in \mathbb{Z}_{\geq 0}$. The upper side and lower side of the six types of indecomposables are interval representations, therefore this decomposition is also the decomposition for each side's $A_3(bf)$ type representations. Since the induced maps are defined by taking $\mathbb{I}[1,3] = 111$, and the elements in HX and HY corresponding to $\begin{smallmatrix}111\\111\end{smallmatrix}$ are the same for each side, the induced maps are the same for each side. □

4.2. Persistence Analysis for Sampled Maps

The ability to decompose and focus exclusively on the interval representation $\mathbb{I}[1,3]$ provides another perspective for analysis of discrete dynamical systems by using eigenspace functors [14]. The paper [14] considers Problem 1.2.6 with the additional assumption that $X = Y$ in order to analyze sampled discrete dynamical systems. Let us reproduce here the problem.

Problem 1.2.6. Let $f\colon X \to Y$ be a continuous map for $X, Y \subset \mathbb{R}^n$. If X, Y, and f are unknown, and we know only a sampled map $f\restriction_S\colon S \to f(S)$ which is a restriction of f on a finite subset $S \subset X$, then can we retrieve any information about the homology induced map $f_*\colon HX \to HY$?

Note that the sampling S is a point cloud capturing topological features of X when S is sufficiently dense.

This section presents construction of an approach aimed at addressing this problem. We explain how to construct the persistent homology of a sampled map, which captures the generator of HX and $Hf(X)$ connected by f. The setup is almost identical to that for the paper [14], until making the following homology sequence (4.2).

4.2.1. Algorithm

First, we generate a filtration of simplicial complexes

$$C_1 \subset C_2 \subset \cdots \subset C_n,$$

each simplex of which has elements of S as its vertices, so that the filtration can capture the topology of the underlying space X. For example, Čech complexes or Vietoris–Rips complexes in Definition 2.2.1 are available. Similarly, we generate a filtration of simplicial complexes

$$D_1 \subset D_2 \subset \cdots \subset D_n$$

for $f(S)$.

Although we expect the sampled map to derive a simplicial map $C_i \to D_i$ on each i-th filter to analyze persistence of the original map f, in general, they can derive only a simplicial partial map[1] $\kappa_i \colon C_i \nrightarrow D_i$. Writing the domain of κ_i as $\operatorname{dom}\kappa_i$, this sequence of the partial maps can be regarded as a sequence $\{\, C_i \xhookleftarrow{\iota_i} \operatorname{dom}\kappa_i \xrightarrow{\kappa_i'} D_i \,\}$ of pairs of maps (ι_i, κ_i'). This sequence of pairs constitutes a filtration induced by the inclusion maps of the filtration

$$\begin{array}{lccccc}
 & \vdots & & \vdots & & \vdots \\
 & \uparrow & & \uparrow & & \uparrow \\
\kappa_{i+1}\colon & C_{i+1} & \xhookleftarrow{\iota_{i+1}} & \operatorname{dom}\kappa_{i+1} & \xrightarrow{\kappa_{i+1}'} & D_{i+1} \\
 & \uparrow & & \uparrow & & \uparrow \\
\kappa_i\colon & C_i & \xhookleftarrow{\iota_i} & \operatorname{dom}\kappa_i & \xrightarrow{\kappa_i'} & D_i \\
 & \uparrow & & \uparrow & & \uparrow \\
 & \vdots & & \vdots & & \vdots
\end{array} .$$

Applying the homology functor to the sequence, we obtain a sequence of representations

[1] A correspondence F from X to Y is *partial map* if $F(x)$ is a singleton or empty set for all $x \in X$, where $F(x) := \{\, y \in Y \mid (x, y) \in F \,\}$.

of the $A_3(bf)$ type quiver as

$$\begin{array}{cccccc} & \vdots & & \vdots & & \vdots \\ & \uparrow & & \uparrow & & \uparrow \\ \kappa_{i+1*}\colon & HC_{i+1} & \xleftarrow{\iota_{i+1*}} & H\operatorname{dom}\kappa_{i+1} & \xrightarrow{\kappa'_{i+1*}} & HD_{i+1} \\ & \uparrow & & \uparrow & & \uparrow \\ \kappa_{i*}\colon & HC_i & \xleftarrow{\iota_{i*}} & H\operatorname{dom}\kappa_i & \xrightarrow{\kappa'_{i*}} & HD_i \\ & \uparrow & & \uparrow & & \uparrow \\ & \vdots & & \vdots & & \vdots \end{array} . \tag{4.2}$$

Here, decomposing each filter $(HC_i \leftarrow H\operatorname{dom}\kappa_i \rightarrow HD_i)$ as a representation, this sequence of representations is isomorphic to

$$\Lambda : \bigoplus_{1\le b\le d\le 3} \mathbb{I}[b,d]^{m^1_{b,d}} \rightarrow \bigoplus_{1\le b\le d\le 3} \mathbb{I}[b,d]^{m^2_{b,d}} \rightarrow \cdots \rightarrow \bigoplus_{1\le b\le d\le 3} \mathbb{I}[b,d]^{m^n_{b,d}}.$$

Projecting to $\mathbb{I}[1,3]$ again yields a sequence of subrepresentations

$$\Lambda[1,3] : \mathbb{I}[1,3]^{m^1_{1,3}} \rightarrow \mathbb{I}[1,3]^{m^2_{1,3}} \rightarrow \cdots \rightarrow \mathbb{I}[1,3]^{m^n_{1,3}}.$$

We must be careful in the construction of $\Lambda[1,3]$. We write canonical projections and injections defined by direct sum respectively as

$$\begin{aligned} \pi_i &\colon \bigoplus_{1\le b\le d\le 3} \mathbb{I}[b,d]^{m^i_{b,d}} \rightarrow \mathbb{I}[1,3]^{m^i_{1,3}} \\ \iota_i &\colon \mathbb{I}[1,3]^{m^i_{1,3}} \rightarrow \bigoplus_{1\le b\le d\le 3} \mathbb{I}[b,d]^{m^i_{b,d}}, \end{aligned}$$

and the morphisms in Λ as

$$\Phi_i \colon \bigoplus_{1\le b\le d\le 3} \mathbb{I}[b,d]^{m^i_{b,d}} \rightarrow \bigoplus_{1\le b\le d\le 3} \mathbb{I}[b,d]^{m^{i+1}_{b,d}}.$$

Then, the morphisms in $\Lambda[1,3]$

$$\Phi_i{}^{1:3}_{1:3} \colon \mathbb{I}[1,3]^{m^i_{1,3}} \rightarrow \mathbb{I}[1,3]^{m^{i+1}_{1,3}}$$

are defined as

$$\Phi_i{}^{1:3}_{1:3} := \pi_{i+1} \circ \Phi_i \circ \iota_i,$$

which is the submatrix at $(1{:}3, 1{:}3)$ in a block matrix form of Φ_i.

At a glance, this construction seems natural, but "$\pi\colon \Lambda \to \Lambda[1,3]$" is not a morphism in the representation category. In other words,

$$\begin{array}{ccc} \bigoplus_{1\le b\le d\le 3} \mathbb{I}[b,d]^{m^i_{b,d}} & \xrightarrow{\Phi_i} & \bigoplus_{1\le b\le d\le 3} \mathbb{I}[b,d]^{m^{i+1}_{b,d}} \\ \downarrow{\scriptstyle \pi_i} & & \downarrow{\scriptstyle \pi_{i+1}} \\ \mathbb{I}[1,3]^{m^i_{1,3}} & \xrightarrow{\Phi_i{}^{1:3}_{1:3}} & \mathbb{I}[1,3]^{m^{i+1}_{1,3}} \end{array}$$

does not always commute. Consequently, the choice of isomorphism of indecomposable decomposition on each filter ($HC_i \leftarrow H\,\mathrm{dom}\,\kappa_i \to HD_i$) might make a difference in the output persistence module, which we will construct later (Diagram (4.3)). In order to make sense of this analysis, the output persistence module should be uniquely determined and independent of the choice of isomorphism. This determination is guaranteed by the following Theorem 4.2.3.

The restriction to the block (1:3, 1:3) has the following functoriality.

Lemma 4.2.1. Let

$$\Phi\colon \bigoplus_{1\le b\le d\le 3} \mathbb{I}[b,d]^{m^1_{b,d}} \to \bigoplus_{1\le b\le d\le 3} \mathbb{I}[b,d]^{m^2_{b,d}}$$

and

$$\Psi\colon \bigoplus_{1\le b\le d\le 3} \mathbb{I}[b,d]^{m^2_{b,d}} \to \bigoplus_{1\le b\le d\le 3} \mathbb{I}[b,d]^{m^3_{b,d}}$$

be block matrix forms, then

$$[\Psi\Phi]^{1:3}_{1:3} = \Psi^{1:3}_{1:3}\Phi^{1:3}_{1:3}.$$

Proof. Let the corresponding scalar matrix symbols of Φ and Ψ be M and N respectively. The block matrix of $\Psi\Phi$ at (1:3, 1:3) is

$$\begin{aligned} [\Psi\Phi]^{1:3}_{1:3} &= \sum_{\mathbb{I}[1,3]\trianglerighteq\mathbb{I}[a,b]\trianglerighteq\mathbb{I}[1,3]} (N^{1:3}_{a:b} f^{1:3}_{a:b})(M^{a:b}_{1:3} f^{a:b}_{1:3}) \\ &= \Big(\sum_{\mathbb{I}[1,3]\trianglerighteq\mathbb{I}[a,b]\trianglerighteq\mathbb{I}[1,3]} N^{1:3}_{a:b} M^{a:b}_{1:3}\Big) f^{1:3}_{1:3} \end{aligned}$$

but only $\mathbb{I}[1,3]$ can be the candidate for interval $\mathbb{I}[a,b]$. Recalling again that $\mathbb{I}[a,b] \not\trianglerighteq \mathbb{I}[c,d]$ is equivalent to $(c{:}d, a{:}b) = \emptyset$, we can check this limitation by the fact that all the right lower blocks of the block matrix form of $CL(bf)$ (presented in Example 3.2.2) are strongly zero. That is, $\mathbb{I}[1,3] \not\trianglerighteq \mathbb{I}[a,b]$ or $\mathbb{I}[a,b] \not\trianglerighteq \mathbb{I}[1,3]$ if $(a,b) \neq (1,3)$. Therefore,

$$\begin{aligned} [\Psi\Phi]^{1:3}_{1:3} &= N^{1:3}_{1:3} M^{1:3}_{1:3} f^{1:3}_{1:3} \\ &= (N^{1:3}_{1:3} f^{1:3}_{1:3})(M^{1:3}_{1:3} f^{1:3}_{1:3}) \\ &= \Psi^{1:3}_{1:3}\Phi^{1:3}_{1:3}. \end{aligned}$$

□

Lemma 4.2.2. Let $\Phi\colon \bigoplus_{1\le b\le d\le 3} \mathbb{I}[b,d]^{m^1_{b,d}} \to \bigoplus_{1\le b\le d\le 3} \mathbb{I}[b,d]^{m^2_{b,d}}$ be the block matrix form of an isomorphism, then $\Phi^{1:3}_{1:3}$ is an isomorphism.

Proof. Let Ψ be the inverse of Φ. By Lemma 4.2.1,

$$[\Psi\Phi]^{1:3}_{1:3} = \Psi^{1:3}_{1:3}\Phi^{1:3}_{1:3}$$

and the left-hand side is the block $(1{:}3, 1{:}3)$ of the identity map, which is the identity map on $\mathbb{I}[1,3]^{m^1_{1,3}}$. Similarly, $\Phi^{1:3}_{1:3}\Psi^{1:3}_{1:3}$ is an identity map on $\mathbb{I}[1,3]^{m^2_{1,3}}$, hence it follows that $\Phi^{1:3}_{1:3}$ is isomorphic. □

Theorem 4.2.3. The isomorphism class of

$$\Phi_i{}^{1:3}_{1:3}\colon \mathbb{I}[1,3]^{m^i_{1,3}} \to \mathbb{I}[1,3]^{m^{i+1}_{1,3}}$$

is uniquely determined and independent of the choice of the basis of

$$\Phi_i\colon \bigoplus_{1\le b\le d\le 3} \mathbb{I}[b,d]^{m^i_{b,d}} \to \bigoplus_{1\le b\le d\le 3} \mathbb{I}[b,d]^{m^{i+1}_{b,d}}.$$

Proof. Let Ψ_i be a morphism isomorphic to Φ_i, which is written as a commutative diagram

$$\begin{array}{ccc} \bigoplus_{1\le b\le d\le n} \mathbb{I}[b,d]^{m_{b,d}} & \xrightarrow{\Phi_i} & \bigoplus_{1\le b\le d\le n} \mathbb{I}[b,d]^{m'_{b,d}} \\ C\downarrow\cong & & R\uparrow\cong \\ \bigoplus_{1\le b\le d\le n} \mathbb{I}[b,d]^{m_{b,d}} & \xrightarrow{\Psi_i} & \bigoplus_{1\le b\le d\le n} \mathbb{I}[b,d]^{m'_{b,d}} \end{array}$$

for some isomorphisms C and R. Namely, $\Phi_i = R\Psi_i C$, and by Lemma 4.2.1, applying the restriction yields

$$\begin{aligned} \Phi_i{}^{1:3}_{1:3} &= [R\Psi_i C]^{1:3}_{1:3} \\ &= R^{1:3}_{1:3}\Psi_i{}^{1:3}_{1:3}C^{1:3}_{1:3}. \end{aligned}$$

By Lemma 4.2.2, $R^{1:3}_{1:3}$ and $C^{1:3}_{1:3}$ are isomorphisms, hence $\Phi_i{}^{1:3}_{1:3}$ is isomorphic to $\Psi_i{}^{1:3}_{1:3}$. □

Since regarding $\mathbb{I}[1,3] = (K \xleftarrow{\mathrm{id}_K} K \xrightarrow{\mathrm{id}_K} K)$ as K omits no information, the sequence $\Lambda[1,3]$ can be seen as a sequence of vector spaces

$$K^{m^1_{1,3}} \to K^{m^2_{1,3}} \to \cdots \to K^{m^n_{1,3}}, \tag{4.3}$$

which is an A_n type representation. We call this representation *persistent homology of the sampled map* $f\restriction_S$. Decomposing into intervals, we can draw a persistence diagram, which shows us the length of the generators of homology in both filtrations which are assigned by f. Simultaneously, we have constructed the filtration of complexes approximating the unknown spaces X and $f(X)$.

This persistence diagram provides no information about eigenvectors, unlike earlier research [14]. However, it can be applied widely. First, since our method does not use the

eigenspace functor, we need not require both sides' spaces to be the same. Therefore, even in the case of sampled dynamical systems $X = Y$ like the earlier research, we can weaken the assumption $f|_S \colon S \to S$ to $f|_S \colon S \to f(S)$ and take another filtration on $f(S)$. (If $f(S)$ is not sufficiently dense for sampling X, then we can take $S \cup f(S)$ instead.) Moreover, since the previous method needs to set an eigenvalue before analysis, they have to predict some behavior of f in advance. By contrast, our method needs no prior information.

4.3. Stability Analysis

For a tool in topological data analysis to be regarded as practical, the output persistence diagrams should behave continuously toward input data. Such a property is known as a stability theorem [10, 8]. Actually, the property has been proved for persistence modules on $\mathbb{R}$.

Let vect be the category of finite dimensional vector spaces, $\mathbb{R}$ be the poset category of real numbers[2]. An object of the functor category $\mathsf{vect}^{\mathbb{R}}$ is also called a persistence module in some papers. To distinguish it from our definition, we call this an $\mathbb{R}$-*persistence module*.

Specifically, for an $\mathbb{R}$-persistence module M, we assign a vector space M_t for $t \in \mathbb{R}$ and a linear map $\varphi_M(s,t) \colon M_s \to M_t$ for $s \leq t \in \mathbb{R}$, where

$$\varphi_M(t,t) = \mathrm{id}_{M_t} \quad \text{and} \quad \varphi_M(s,t) \circ \varphi_M(r,s) = \varphi_M(r,t)$$

for all $r \leq s \leq t \in \mathbb{R}$. We designate the linear maps $\varphi_M(s,t)$ *transition maps*. A morphism $f \colon M \to N$ of $\mathbb{R}$-persistence modules is a natural transformation: a collection of morphisms $\{ f_t \colon M_t \to N_t \mid t \in \mathbb{R} \}$ commuting

$$\begin{array}{ccc} M_s & \xrightarrow{\varphi_M(s,t)} & M_t \\ \downarrow{\scriptstyle f_s} & & \downarrow{\scriptstyle f_t} \\ N_s & \xrightarrow{\varphi_N(s,t)} & N_t \end{array}$$

for all $s \leq t \in \mathbb{R}$.

It is noteworthy that every persistence module can be similarly regarded as a functor from a finite poset category to vect. Moreover, every persistent homology (2.3) can be embedded into an $\mathbb{R}$-persistence module.

The fundamental objects of $\mathbb{R}$-persistence modules are *interval modules* K_I for intervals $I \subset \mathbb{R}$, given as $(K_I)_t = K$ for $t \in I$ and $(K_I)_t = 0$ otherwise, and with the morphism corresponding to $s \leq t \in I$ is an identity map. As is the case with persistent homology, every $\mathbb{R}$-persistence module can be decomposed into a direct sum of interval modules [11].

We can define a distance between $\mathbb{R}$-persistence modules, called the interleaving distance.

[2]For $x, y \in \mathbb{R}$, a morphism $x \to y$ uniquely exists if and only if $x \leq y$.

Definition 4.3.1. For $\delta \geq 0$, define the functor $(\cdot)(\delta)\colon \mathsf{vect}^{\mathbb{R}} \to \mathsf{vect}^{\mathbb{R}}$, called the *shift functor*, as follows. For an $\mathbb{R}$-persistence module M, $M(\delta)_t := M_{t+\delta}$ and $\varphi_{M(\delta)}(s,t) := \varphi_M(s+\delta, t+\delta)$. For a morphism f in $\mathsf{vect}^{\mathbb{R}}$, $f(\delta) := f_{t+\delta}$.

Definition 4.3.2. For an $\mathbb{R}$-persistence module M and $\delta \geq 0$, the *δ-transition morphism* $\varphi_M(\delta)\colon M \to M(\delta)$ is defined as $\varphi_M(\delta)_t := \varphi_M(t, t+\delta)$.

Definition 4.3.3. $\mathbb{R}$-persistence modules M and N are said to be *δ-interleaved* if there exist morphisms $f\colon M \to N(\delta)$ and $g\colon N \to M(\delta)$ such that

$$g(\delta) \circ f = \varphi_M(2\delta) \quad \text{and} \quad f(\delta) \circ g = \varphi_N(2\delta).$$

The *interleaving distance* $d_I\colon \mathsf{vect}^{\mathbb{R}} \times \mathsf{vect}^{\mathbb{R}} \to [0, \infty]$ is defined as

$$d_I(M,N) := \inf_{\delta} \{\, M \text{ and } N \text{ are } \delta\text{-interleaved} \,\}.$$

An often used distance between persistence diagrams is the *bottleneck distance*, which is defined by bijections between them. It is well-known that the interleaving distance of $\mathbb{R}$-persistence modules is equal to the bottleneck distance of their persistence diagrams [23, 3]. Consequently, by showing that a distance between input data is greater than the interleaving distance of their $\mathbb{R}$-persistence module, we can prove the stability of the persistence diagrams toward input data.

In analogy with the paper of eigenspace functors [14], stability theorems for some filtrations also hold on our analysis as follows. The discrete setting discussed in Subsection 4.2.1 is sufficient for implementation, but we extend it to a continuous analysis to prove its stability.

Now we use the following filtrations for S and $f(S)$. Let $d_{\mathbb{R}^n \times \mathbb{R}^n}$ be a distance on $\mathbb{R}^n \times \mathbb{R}^n$ defined as

$$d_{\mathbb{R}^n \times \mathbb{R}^n}((x_1,y_1),(x_2,y_2)) := \max\{d_{\mathbb{R}^n}(x_1,x_2), d_{\mathbb{R}^n}(y_1,y_2)\},$$

where $d_{\mathbb{R}^n}$ is the Euclidean metric on $\mathbb{R}^n$. For a subset B of $\mathbb{R}^n$, we define a function $d_B\colon \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ to be infimum distance to a point in B. Similarly, we abuse the same symbol and define the function $d_B\colon \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ for a subset B of another space $\mathbb{R}^n \times \mathbb{R}^n$. We use the notation $B_r := d_B^{-1}[0,r]$ to denote the sublevel sets.

Let $\mathsf{Top}^{(\mathsf{bf})}$ be the functor category from the $A_3(bf)$ type quiver $(\cdot \leftarrow \cdot \rightarrow \cdot)$ as a poset category to the category of topological spaces. The sublevel sets S_r, $f(S)_r$, and $\mathrm{Gr}(f\restriction_S)_r$ constitute the filtration $\{\, S_r \leftarrow \mathrm{Gr}(f\restriction_S)_r \rightarrow f(S)_r \,\}$ in $\mathsf{Top}^{(\mathsf{bf})}$ with morphisms induced by the inclusions which commute the diagram

$$\begin{array}{ccccc} S_r & \longleftarrow & \mathrm{Gr}(f\restriction_S)_r & \longrightarrow & f(S)_r \\ \uparrow & & \uparrow & & \uparrow \\ S_s & \longleftarrow & \mathrm{Gr}(f\restriction_S)_s & \longrightarrow & f(S)_s \end{array} \tag{4.4}$$

for every $s \leq r \in \mathbb{R}_{\geq 0}$.

In the same way as that for the discrete analysis, applying the homology functor H to the filtration produces $\{ HS_r \leftarrow H\,\mathrm{Gr}(f\restriction_S)_r \rightarrow Hf(S)_r \}$, which is a family of objects in the representation category $\mathrm{rep}(A_3(bf))$ with the induced morphisms from Diagram (4.4).

Remark. We have constructed the different representations from the previous representations using complexes and domains, but these are isomorphic if we adopt Čech complexes. It is known by the Nerve Lemma [4] that, if B is a finite subset in a metric space, then the sublevel set B_r is homotopy equivalent to the Čech complex of B with radius r. Therefore, letting C_r, E_r, and D_r be Čech complexes with radius r of the finite subsets S, $\mathrm{Gr}(f\restriction_S)$, and $f(S)$ respectively, this family is isomorphic to the family $\{ HC_r \leftarrow HE_r \rightarrow HD_r \}$. Moreover, the Projection Lemma [14, p. 1231] shows that this family is isomorphic to the family $\{ HC_r \leftarrow H\,\mathrm{dom}\,\kappa_r \rightarrow HD_r \}$.

Since decomposing every representation into intervals is isomorphic in the functor category $\mathrm{rep}(A_3(bf))^{\mathbb{R}}$, the family $\{ HS_r \leftarrow H\,\mathrm{Gr}(f\restriction_S)_r \rightarrow Hf(S)_r \}$ is isomorphic to $\{ \bigoplus_{1\le b\le d\le 3} \mathbb{I}[b,d]^{m^r_{b,d}} \}$, and the induced morphisms can be written in block matrix form again.

By Lemma 4.2.1 and Theorem 4.2.3, the family $\{ \mathbb{I}[1,3]^{m^r_{1,3}} \}$ and the induced morphisms are uniquely determined up to isomorphism. This gives us three copies of the $\mathbb{R}$-persistence module $\{ K^{m^r_{1,3}} = K^{m^r_{1,3}} = K^{m^r_{1,3}} \}$. Thus, we obtain an $\mathbb{R}$-persistence module $\{ K^{m^r_{1,3}} \}$. We denote this $\mathbb{R}$-persistence module of the sampled map $f\restriction_S$ as $M^{f\restriction_S}$ and call it the $\mathbb{R}$-*persistence module of the sampled map.*

Theorem 4.3.4. Let d_H be a Hausdorff distance induced by $d_{\mathbb{R}^n\times\mathbb{R}^n}$. For two sampled maps $h\colon S \to \mathbb{R}^n$ and $h'\colon S' \to \mathbb{R}^n$, let M^h, $M^{h'}$ be the $\mathbb{R}$-persistence modules of the sampled maps. Then,

$$d_I(M^h, M^{h'}) \le d_H(\mathrm{Gr}(h), \mathrm{Gr}(h')).$$

Proof. Let $\varepsilon := d_H(\mathrm{Gr}(h), \mathrm{Gr}(h'))$, and r be an arbitrary non-negative real number.

By the definition of Hausdorff distance, $\mathrm{Gr}(h)_r \subset \mathrm{Gr}(h')_{r+\varepsilon}$ and $\mathrm{Gr}(h')_r \subset \mathrm{Gr}(h)_{r+\varepsilon}$. Moreover, $\varepsilon = d_H(\mathrm{Gr}(h), \mathrm{Gr}(h'))$ implies that $d_H(S, S') \le \varepsilon$ and $d_H(h(S), h'(S')) \le \varepsilon$, hence $S_r \subset S'_{r+\varepsilon}$, $S'_r \subset S_{r+\varepsilon}$, $h(S)_r \subset h'(S')_{r+\varepsilon}$, and $h'(S')_r \subset h(S)_{r+\varepsilon}$ as well. These inclusions lead to the following commutative diagrams:

$$\begin{array}{ccccc} S'_{r+\varepsilon} & \longleftarrow & \mathrm{Gr}(h')_{r+\varepsilon} & \longrightarrow & h'(S')_{r+\varepsilon} \\ \uparrow & & \uparrow & & \uparrow \\ S_r & \longleftarrow & \mathrm{Gr}(h)_r & \longrightarrow & h(S)_r \end{array}$$

and

$$\begin{array}{ccccc} S_{r+\varepsilon} & \longleftarrow & \mathrm{Gr}(h)_{r+\varepsilon} & \longrightarrow & h(S)_{r+\varepsilon} \\ \uparrow & & \uparrow & & \uparrow \\ S'_r & \longleftarrow & \mathrm{Gr}(h')_r & \longrightarrow & h'(S')_r \end{array}.$$

By that functoriality, it is straightforward that these inclusions induce ε-interleaving morphisms between M^h and $M^{h'}$. □

Accordingly, the obtained persistence modules and persistence diagrams can only have as much noise as S or its evaluation by f.

4.4. Discussion

4.4.1. Grid Filtration

In Section 4.3, we observed that the algorithms discussed in Subsection 4.2.1 are applicable to the filtrations generated by expanding a graph. This perspective provides a persistent approach to correspondences of sampled maps constructed by dividing the space [21].

Suppose the spaces X and Y are embedded into Euclidean space $\mathbb{R}^n$, and both $\mathbb{R}^n$ are divided by n-dimensional ε-cubes

$$\{\, [a_1\varepsilon, (a_1+1)\varepsilon] \times \cdots \times [a_n\varepsilon, (a_n+1)\varepsilon] \mid a_1, \ldots, a_n \in \mathbb{Z} \,\}\,.$$

To distinguish the two divisions, we write this set as $\mathcal{X}$ for the X side Euclidean space and $\mathcal{Y}$ for Y side. Let $f\restriction_S$ be a sampled map of a continuous map $f\colon X \to Y$, and p and q be the canonical projections of $\mathbb{R}^n \times \mathbb{R}^n$ to the X-side and Y-side Euclidean spaces respectively. First, we generate a correspondence

$$F_\varepsilon^{f\restriction_S} := \{\, (x,y) \in \mathbb{R}^n \times \mathbb{R}^n \mid x \in \exists X' \in \mathcal{X}, y \in \exists Y' \in \mathcal{Y}, (X' \times Y') \cap \mathrm{Gr}(f\restriction_S) \neq \emptyset \,\}$$

where $\mathrm{Gr}(f\restriction_S) := \{\, (s, f(s)) \mid s \in S \,\}$.

In this setup, we use the L^∞ metric $d_\infty((x_i),(y_i)) := \max_i(|x_i - y_i|)$ for both spaces $\mathbb{R}^n$ and $\mathbb{R}^n \times \mathbb{R}^n$. Defining the filtration of a correspondence

$$F_{i\varepsilon} := (F_\varepsilon^{f\restriction_S})_{i\varepsilon} = \{\, r \in \mathbb{R}^n \times \mathbb{R}^n \mid d_\infty(r, F_\varepsilon^{f\restriction_S}) \leq i\varepsilon \,\} \quad (i \in \mathbb{Z}_{\geq 0})$$

and morphisms $p_{i\varepsilon} := p\restriction_{F_{i\varepsilon}}$ and $q_{i\varepsilon} := q\restriction_{F_{i\varepsilon}}$, we have the similar diagram to that obtained before,

$$\begin{array}{ccccc}
\vdots & & \vdots & & \vdots \\
\uparrow & & \uparrow & & \uparrow \\
p(F_{(i+1)\varepsilon}) & \xleftarrow{p_{(i+1)\varepsilon}} & F_{(i+1)\varepsilon} & \xrightarrow{q_{(i+1)\varepsilon}} & q(F_{(i+1)\varepsilon}) \\
\uparrow & & \uparrow & & \uparrow \\
p(F_{i\varepsilon}) & \xleftarrow{p_{i\varepsilon}} & F_{i\varepsilon} & \xrightarrow{q_{i\varepsilon}} & q(F_{i\varepsilon}) \\
\uparrow & & \uparrow & & \uparrow \\
\vdots & & \vdots & & \vdots
\end{array}$$

thereby allowing us to obtain the filtrations $\{\, p(F_{i\varepsilon}) \,\}$ and $\{\, q(F_{i\varepsilon}) \,\}$, capturing the persistent topological features X and $f(X)$. Again, the homology functor derives the sequence

of morphisms in $\mathrm{rep}(A_3(bf))$. Therefore, we can execute the same analysis as before, transforming it into block matrix form, restricting to the blocks $(1{:}3, 1{:}3)$, identifying it with a representation of the quiver A_n, and consequently producing a persistence diagram.

This method of analysis has the following stability. We remark that the output persistent homology is regarded as an $\mathbb{R}$-persistence module by the embedding induced by

$$F_r := \begin{cases} 0 & r < 0 \\ F_{i\varepsilon} & i\varepsilon \leq r < (i+1)\varepsilon \text{ for } i \in \mathbb{Z}_{\geq 0} \end{cases}.$$

Theorem 4.4.1. Let d_H be a Hausdorff distance induced by d_∞. For two sampled maps $h\colon S \to \mathbb{R}^n$ and $h'\colon S' \to \mathbb{R}^n$, let $\{ F_{i\varepsilon} \}$ and $\{ F'_{i\varepsilon} \}$ be the respective filtrations of correspondences, and let M^h and $M^{h'}$ be their output $\mathbb{R}$-persistence modules. If $d_H(\mathrm{Gr}(h), \mathrm{Gr}(h')) \leq \varepsilon$, then $d_I(M^h, M^{h'}) \leq \varepsilon$.

Proof. The assumption $d_H(\mathrm{Gr}(h), \mathrm{Gr}(h')) \leq \varepsilon$ derives the inequality $d_H(F^h_\varepsilon, F^{h'}_\varepsilon) \leq \varepsilon$. Therefore there exist the inclusions

$$\begin{array}{ccccc} p(F'_{(i+1)\varepsilon}) & \longleftarrow & F'_{(i+1)\varepsilon} & \longrightarrow & q(F'_{(i+1)\varepsilon}) \\ \uparrow & & \uparrow & & \uparrow \\ p(F_{i\varepsilon}) & \longleftarrow & F_{i\varepsilon} & \longrightarrow & q(F_{i\varepsilon}) \end{array}$$

and

$$\begin{array}{ccccc} p(F_{(i+1)\varepsilon}) & \longleftarrow & F_{(i+1)\varepsilon} & \longrightarrow & q(F_{(i+1)\varepsilon}) \\ \uparrow & & \uparrow & & \uparrow \\ p(F'_{i\varepsilon}) & \longleftarrow & F'_{i\varepsilon} & \longrightarrow & q(F'_{i\varepsilon}) \end{array}.$$

It is clear that these inclusions induce the ε-interleaving morphisms between M^h and $M^{h'}$. □

4.4.2. Functoriality for Other Intervals and Orientations

The functoriality lemma, Lemma 4.2.1, can be generalized for the restriction to every "diagonal" block. Precisely, since the candidate of intervals $\mathbb{I}[c, d]$ satisfying relations $\mathbb{I}[a, b] \trianglerighteq \mathbb{I}[c, d] \trianglerighteq \mathbb{I}[a, b]$ is only $\mathbb{I}[c, d] = \mathbb{I}[a, b]$,

$$[\Psi\Phi]^{a:b}_{a:b} = \Psi^{a:b}_{a:b}\Phi^{a:b}_{a:b}.$$

holds for all $\mathbb{I}[a, b]$. This result can be checked easily, not only on the orientation bf but also on every orientation of any length, as follows.

Suppose $\mathbb{I}[c, d] \neq \mathbb{I}[a, b]$, which can happen when $a \neq c$ or when $b \neq d$. In the case that $a \neq c$, we may assume $a < c$ without loss of generality. When $(c-1)$-th orientation

is f, consider $g = \{ g_i \}_{i=1}^{n} \in \mathrm{Hom}(\mathbb{I}[a,b], \mathbb{I}[c,d])$. Then, the commutative diagram of the morphism g from $(c-1)$ to c is

$$\begin{array}{ccc} 0 & \xrightarrow{\mathrm{id}_K} & K \\ {\scriptstyle g_{c-1}}\uparrow & & \uparrow{\scriptstyle g_c} \\ K & \xrightarrow{\mathrm{id}_K} & K \end{array} .$$

It is obvious that $g_{c-1} = 0$, and the commutativity derives $g_c = 0$. By the same discussion in the proof of Lemma 3.1.3, the commutativity of the diagram on g derives $g_i = 0$ for the other vertices i, i.e., $g = 0$. Consequently, $\mathrm{Hom}(\mathbb{I}[a,b], \mathbb{I}[c,d]) = 0$, hence $\mathbb{I}[a,b] \not\trianglerighteq \mathbb{I}[c,d]$. We can show $\mathbb{I}[c,d] \not\trianglerighteq \mathbb{I}[a,b]$ when $(c-1)$-th orientation is b in a similar discussion, using the commutative diagram

$$\begin{array}{ccc} K & \xleftarrow{\mathrm{id}_K} & K \\ {\scriptstyle g_{c-1}}\uparrow & & \uparrow{\scriptstyle g_c} \\ 0 & \xleftarrow{\mathrm{id}_K} & K \end{array} .$$

Similar arguments also hold in the case that $b \neq d$, concluding $\mathbb{I}[a,b] \not\trianglerighteq \mathbb{I}[c,d]$ or $\mathbb{I}[c,d] \not\trianglerighteq \mathbb{I}[a,b]$.

That is why we can extend the statement on the orientation bf to general τ_n as shown below.

Proposition 4.4.2. Let

$$\Phi : \bigoplus_{1 \le a \le b \le n} \mathbb{I}[a,b]^{m^1_{a,b}} \to \bigoplus_{1 \le a \le b \le n} \mathbb{I}[a,b]^{m^2_{a,b}}$$

and

$$\Psi : \bigoplus_{1 \le a \le b \le n} \mathbb{I}[a,b]^{m^2_{a,b}} \to \bigoplus_{1 \le a \le b \le n} \mathbb{I}[a,b]^{m^3_{a,b}}$$

be block matrix forms of objects in the arrow category $\mathrm{arr}(\mathrm{rep}(A_n(\tau_n)))$, then

$$[\Psi\Phi]^{a:b}_{a:b} = \Psi^{a:b}_{a:b}\Phi^{a:b}_{a:b}$$

for all $1 \le a \le b \le n$.

This property can be utilized for *2-D persistence modules*, which are representations with the shape

$$\begin{array}{ccccccc} M_{n_1,1} & \longleftrightarrow & M_{n_1,2} & \longleftrightarrow & \cdots & \longleftrightarrow & M_{n_1,n_2} \\ \uparrow & & \uparrow & & & & \uparrow \\ \vdots & & \vdots & & & & \vdots \\ \uparrow & & \uparrow & & & & \uparrow \\ M_{2,1} & \longleftrightarrow & M_{2,2} & \longleftrightarrow & \cdots & \longleftrightarrow & M_{2,n_2} \\ \uparrow & & \uparrow & & & & \uparrow \\ M_{1,1} & \longleftrightarrow & M_{1,2} & \longleftrightarrow & \cdots & \longleftrightarrow & M_{1,n_2} \end{array} \tag{4.5}$$

where every row has the same orientation τ_{n_2}. The 2-D persistence modules sometimes appear and cause difficulties in the context of persistence analysis for time series data. See [7] for details and higher dimensional persistence.

In our context, the 2-D persistence module naturally appears when we consider iterations of a sampled map or compositions of sampled maps. Suppose we have a time series of some point clouds $\{ S_1, S_2, \ldots, S_T \}$ in the same Euclidean space, with their transition as maps $\{ f_i \colon S_i \to S_{i+1} \}$. We generate a filtration of simplicial complexes $C_1^t \subset \cdots \subset C_n^t$ for each S_t. As shown in Subsection 4.2.1, the maps between points induce a filtration of partial maps $\kappa_i^t \colon C_i^t \nrightarrow C_i^{t+1}$, which induces a commutative diagram

$$
\begin{array}{ccccccc}
\vdots & & \vdots & & \vdots & & \vdots \\
\uparrow & & \uparrow & & \uparrow & & \uparrow \\
C_{i+1}^1 & \hookleftarrow & \operatorname{dom} \kappa_{i+1}^1 & \longrightarrow & C_{i+1}^2 & \hookleftarrow \cdots \longrightarrow & C_{i+1}^T \\
\uparrow & & \uparrow & & \uparrow & & \uparrow \\
C_i^1 & \hookleftarrow & \operatorname{dom} \kappa_i^1 & \longrightarrow & C_i^2 & \hookleftarrow \cdots \longrightarrow & C_i^T \\
\uparrow & & \uparrow & & \uparrow & & \uparrow \\
\vdots & & \vdots & & \vdots & & \vdots
\end{array}
\tag{4.6}
$$

by taking their domains. As a consequence, the homology functor induces the 2-D persistence module from the above diagram. In this case, we can observe the 2-D persistence module from the viewpoint that the horizontal (vertical) direction on the diagram describes persistence in time (space, respectively).

Let us go back to Diagram (4.5). In the same way as in the specific case $\tau_{n_2} = bf$, Diagram (4.5) can be regarded as a sequence of morphisms in the category $\mathrm{rep}(A_{n_2}(\tau_{n_2}))$. By decomposing representations of $A_{n_2}(\tau_{n_2})$ in each row, we can deal with the morphisms as matrices in block matrix form. Restricting each matrix to the diagonal block $(a{:}b, a{:}b)$ derives a sequence of matrices whose domains and codomains are direct sums of $\mathbb{I}[a, b]$. As this sequence comprises $b-a$ copies of nonzero representations of A_{n_1} and $n_2-(b-a)$ copies of zero representations, we can take one of the nonzero representations. Finally, we obtain the persistence diagram by decomposition. Proposition 4.4.2 ensures the uniqueness of the output persistence diagram.

In the case of 2-D persistence modules derived from Diagram (4.6), when we take the block $(a{:}b, a{:}b)$ as $(1{:}n_2, 1{:}n_2)$, each generator of the output persistence module survives under all transitions, and its lifetime in the persistence diagram shows how robust it is in the Euclidean space. Although this process ignores much information stored in the other blocks, it is an approach to 2-D persistence analysis that can capture the rough topological structures.

4.5. Implementation and Numerical Experiments

The author has implemented the algorithm in Subsection 4.2.1 using the `C++` language. Here we fix the field for the coefficient of matrices and the homology functor as $\mathbb{Z}/1009\mathbb{Z}$.

Remark. To implement the algorithm on computers, we must use finite fields as the coefficient. Here, every map is written as a matrix. If we choose the field $\mathbb{Z}/2\mathbb{Z}$ as the coefficient, then every entry with the prime factor 2 in the matrix is regarded as 0. For example, a homology generator a mapped as $f_*(a) = 2a$, such as the example discussed later and in Figure 4.2, is ignored. Therefore, it is better to choose a larger prime number p as the coefficient $\mathbb{Z}/p\mathbb{Z}$ to retrieve more generators.

The implementation uses Vietoris–Rips complexes for simplicity, although Čech complexes are theoretically more satisfying as mentioned in Subsection 2.2.1. Except for the construction of the persistence module from the sequence of the pairs of the maps $\{(\iota_{i*}, \kappa'_{i*})\}$, it basically follows the algorithm in [14].

First, we generate the boundary matrix induced by the filtration of the Vietoris–Rips complexes for each point clouds S and $f(S)$; then we generate the boundary matrix of the domain filtration. These boundary matrices are stored as sparse matrices of the `Eigen` library [30], and we use the original persistence algorithm [13] to compute the reduced boundary matrices and the bases of the persistent homology of the filtration.

Second, since we can obtain the homology bases for each filtration, we generate the maps ι_{i*} and κ'_{i*} as matrices between the homology basis for each filter. In the same way, we compute the induced maps of the inclusions $j_*\colon H \operatorname{dom} \kappa_i \to H \operatorname{dom} \kappa_{i+1}$ as matrices. To obtain the basis of $\mathbb{I}[1,3]$ for each matrix ι_{i*} and κ'_{i*}, we execute the following elementary row and column operations.

1. Transform ι_{i*} to Smith normal form:
$$P_1 \iota_{i*} Q_1 = \begin{bmatrix} E_{r_1} & 0 \\ 0 & 0 \end{bmatrix},$$
where P_1 and Q_1 respectively denote regular matrices corresponding to elementary operations, and where r_1 is the rank of ι_{i*}.
2. Since ι_{i*} and κ'_{i*} share the same basis for the columns, the elementary column operations Q_1 are performed simultaneously on κ'_{i*}
$$\kappa'_{i*} Q_1 = \begin{bmatrix} X_1 & X_2 \end{bmatrix},$$
where X_1 is the submatrix on the basis of columns corresponding to the above E_{r_1}, and where X_2 is the submatrix corresponding to the 0 columns in $P_1 \iota_{i*} Q_1$.
3. Transform X_2 to Smith normal form with elementary row operations P_2 and elementary column operations Q_2
$$\begin{bmatrix} P_2 X_1 & E_{r_2} & 0 \\ & 0 & 0 \end{bmatrix} = \begin{bmatrix} X_3 & E_{r_2} & 0 \\ X_4 & 0 & 0 \end{bmatrix}$$

where $P_2 X_1$ is divided into submatrices of appropriate sizes on the right-hand side. We remark that these column operations have no side effect on the ι_{i*} side matrix since every column corresponding to the basis is zero.

4. Zero out X_3 by E_{r_2} using column operations without any side effect:

$$\begin{bmatrix} 0 & E_{r_2} & 0 \\ X_4 & 0 & 0 \end{bmatrix}.$$

5. Transform X_4 to Smith normal form:

$$\begin{bmatrix} 0 & & E_{r_2} & 0 \\ E_{r_3} & 0 & 0 & 0 \\ 0 & 0 & & \end{bmatrix}.$$

Here the column operations have a side effect on E_{r_1} transforming it to a matrix X_4, but X_4 is regular, hence we can transform it to E_{r_1} again using only row operations.

6. Finally, we obtain the matrix transformations

$$\iota_{i*} \mapsto \begin{bmatrix} E_{r_3} & 0 & 0 & 0 \\ 0 & E_{r_3 - r_1} & 0 & 0 \end{bmatrix} \text{ and } \kappa'_{i*} \mapsto \begin{bmatrix} 0 & 0 & E_{r_2} & 0 \\ E_{r_3} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

which are decomposed into intervals. The rows and columns corresponding to two E_{r_3} are pairs of identity maps, which are $\mathbb{I}[1,3]$. Therefore the basis in column E_{r_3} is what we want.

Applying the change of basis of $H \operatorname{dom} \kappa_i$ during the above column operations and restricting to the basis corresponding to E_{r_3}, we finally obtain the persistent homology of the sampled map. At last we decompose the persistent homology into intervals using the decomposition algorithm in [14, Subsection 3.4] and plot a persistence diagram.

As an example for input data, let us consider the twice map on the unit circle $f\colon S^1 \to S^1$ defined by $f(z) := z^2$. The sampled points of the unit circle are 100 points $z_j := \cos(2\pi \frac{j}{100}) + \sqrt{-1}\sin(2\pi \frac{j}{100})$ for $0 \leq j < 100$, with added Gaussian noise with $\sigma \in [0.00, 0.30]$.

A computational result is presented in Figure 4.2, which portrays the sampled map at $\sigma = 0.18$ and its unique generator of the persistence diagram. The generator is indeed approximating the unit circle, and we can see that its image is turning around the origin twice. Results for other noises are shown in Figure 4.3.

Figure 4.4 presents the persistence diagrams under changing σ from 0 to 0.3. As expected, the lifetime of the unique generator decreases as the noise gets stronger.

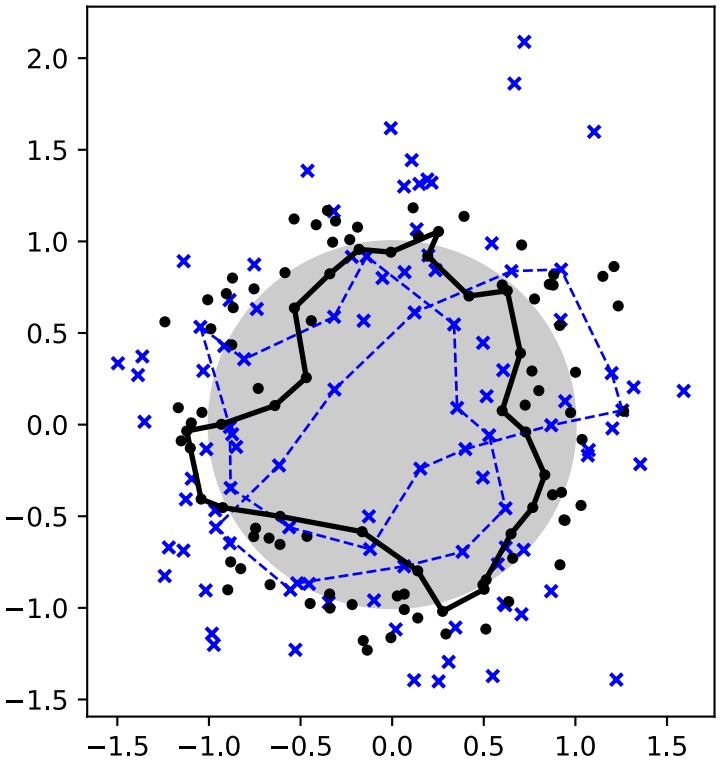


Figure 4.2.: A computational result for $f(z) = z^2$. The number of points is 100 and the Gaussian noise is at $\sigma = 0.18$. The black points are sampled points for the domain and the blue crosses are its image by f. As presented in Figure 4.4, the generator of the persistence diagram of these points is unique, which is described by the black edges approximating the unit circle. The image of the generator is the blue dashed edges, and we can observe that it turns around the origin twice.

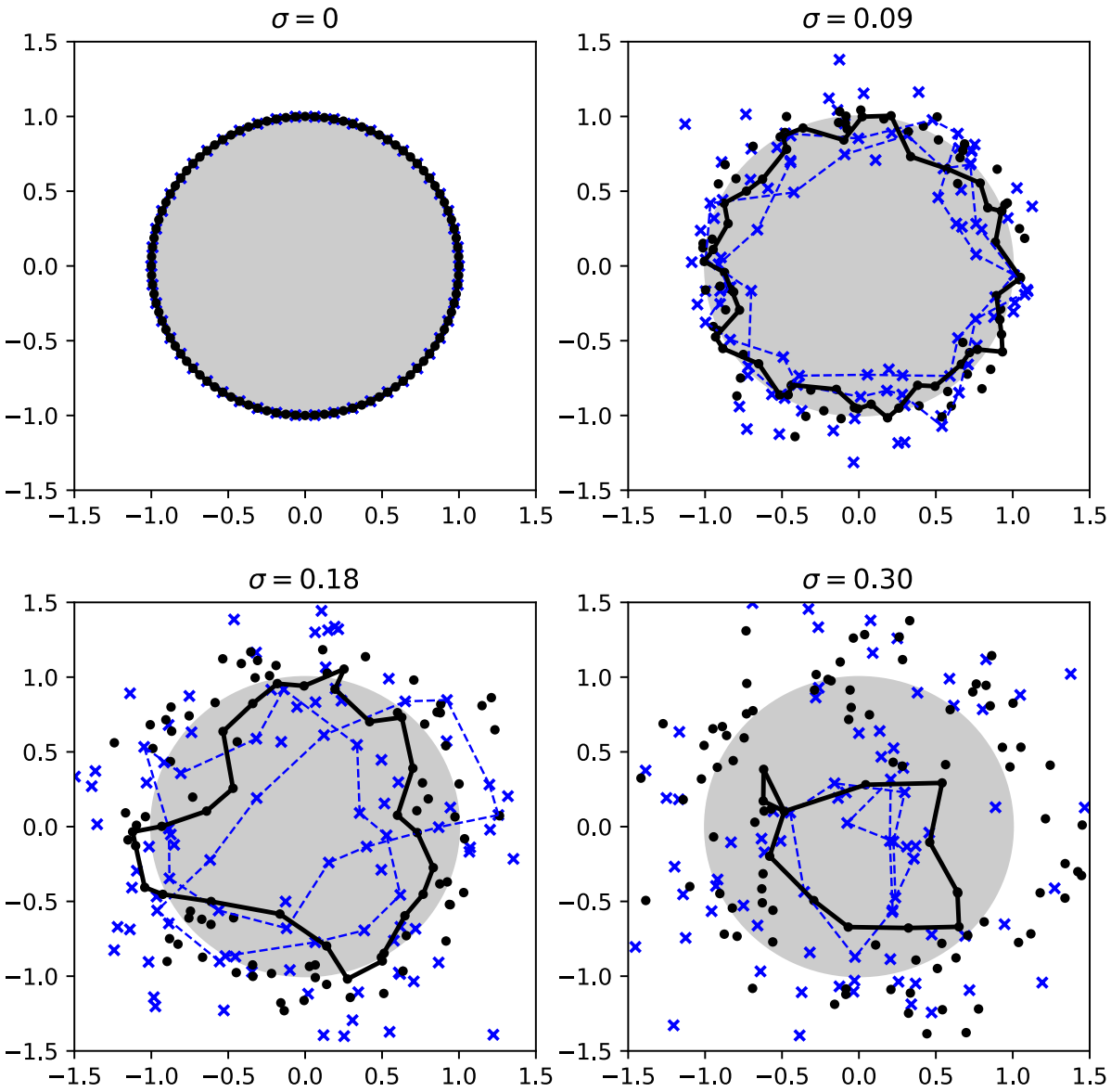


Figure 4.3.: Computational results for $f(z) = z^2$ with 100 points at $\sigma = 0.00$, 0.09, 0.18, and 0.30. Every plotting range is restricted to $[-1.5, 1.5] \times [-1.5, 1.5]$ to observe the generators. The corresponding persistence diagrams are in Figure 4.4.

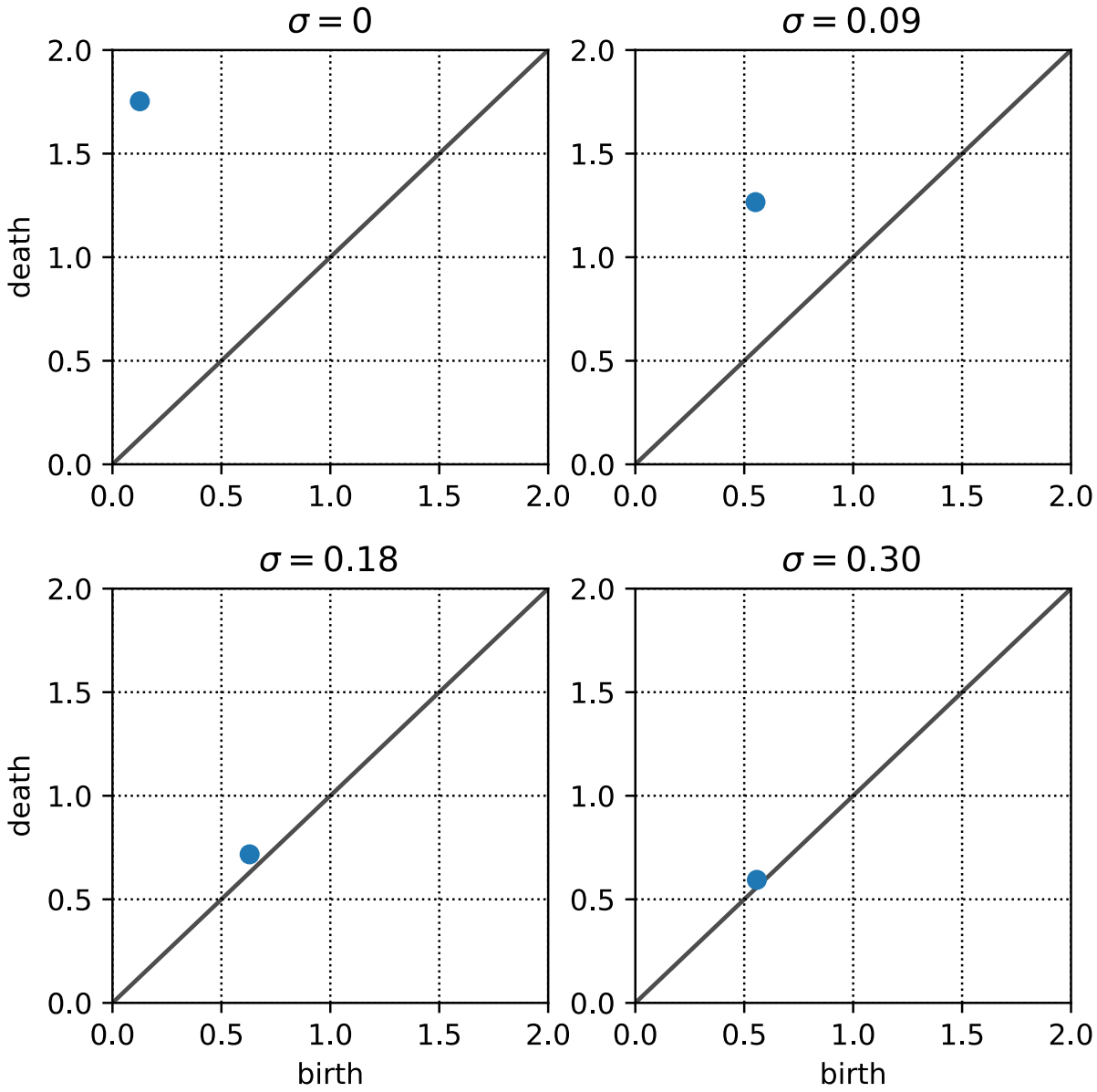


Figure 4.4.: The persistence diagrams for $f(z) = z^2$ with 100 points at $\sigma = 0.00$, 0.09, 0.18, and 0.30. We can observe that each persistence diagram has the unique point, and it goes to the diagonal line as the noise gets stronger.

Appendix A.

Topological Data Analysis on Granular Materials

This appendix is a summary of the mathematical part of the paper:

[26] M. Saadatfar, H. Takeuchi, V. Robins, N. Francois, and Y. Hiraoka, Pore Configuration Landscape of Granular Crystallization, Nat. Commun. 8 (2017), 15082.

This paper demonstrates the usability of persistent homology to investigate configurations of cavities in granular materials using persistent homology. The aim of this appendix is to understand why topological analysis of deformations of point clouds, which was a motivation for both Chapters 3 and 4, is important and needed for experimental data analysis.

A.1. Persistence Diagrams of Granular Packings

Granular packings are phenomena which frequently appear in nature and industrial fields, therefore it is important to study their structure and behaviors. As a simplest model of granular packings, sphere packings have been studied, which are clusters of monosized spheres in containers. Due to improvements of X-ray computed tomography (CT) scans, we are now able to detect more accurately the coordinates of the centers and the radii of such spheres, and research using real data of sphere packings to study packing density, physical property, and geometrical structure of the sphere configurations has begun (for example, see [17]).

In Figure A.1, we can see a result of CT scans of sphere packings, whose data set is voxel data with the size of each voxel $\approx 30\,\mu\text{m}$. It is known that local crystallized structures appear at a density beyond $\phi \approx 0.64$ (see [17]), hence the packing in Figure A.1 consists of amorphous structure and crystallized structure, called a *partially crystallized packing*. From this voxel data, we have computed the center and the diameter for each sphere.

The diameters of spheres are 1 mm with poly-dispersity = 0.025 mm (namely, the width of the distribution of grain diameters is 0.05 mm), and the *density* is defined as the total volume of spheres divided by the volume of cylinder[1].

[1]The height of the cylinder is defined as the upper surface of the spheres.

Figure A.1.: 3D rendering of experimental packings in a cylindrical container with a diameter and height of 66 mm. The number of spheres is 216,722 and the density is 0.69. CT scans enable us to observe the inner structure of granular materials, and we can indeed observe the crystallized structure and amorphous structure. (This figure is subfigure of "3D rendering of spherical and cylindrical experimental packings" by M. Saadatfar et al. `https://www.nature.com/articles/ncomms15082/figures/1`, used under CC BY 4.0 `https://creativecommons.org/licenses/by/4.0/`.)

Consequently, we have obtained a point cloud, where each point has "weight", i.e. the radius. Let $\{(x_i, r_i)\}$ be the data set of the centers x_i and the radii r_i. Then, the packing state can be expressed as $X = \bigcup_i B(x_i; r_i)$, where $B(x_i; r_i)$ is the ball of center x_i and radius r_i. An often used construction of a filtration from such a point cloud with weight is the expansion of balls with the following parameter.

We introduce one parameter $\alpha \in \mathbb{R}$, and define $r_i(\alpha) := \sqrt{r_i^2 + \alpha}$ and $X(\alpha) := \bigcup_i B(x_i; r_i(\alpha))$. For an increasing sequence $0 = \alpha_1 < \alpha_2 < \cdots < \alpha_n$, we obtain a filtration of spaces

$$X(\alpha_1) \subset X(\alpha_2) \subset \cdots \subset X(\alpha_n).$$

In this study, we adopt the 2nd homology functor $H := H_2(-; K)$ with the coefficient $K = \mathbb{Z}/2\mathbb{Z}$, in order to extract the cavity configurations. We thus obtain the persistent homology

$$HX(\alpha_1) \to HX(\alpha_2) \to \cdots \to HX(\alpha_n). \tag{A.1}$$

It is known that each space $X(\alpha_i)$ is homotopy equivalent to a *weighted alpha complex* of the centers [12]. The (weighted) alpha complex is a simplicial complex which is defined by Delaunay triangulation of the centers, and is available in the `C++` library `CGAL` [29].

Finally, we decompose the persistent homology (A.1) into intervals

$$(HX(\alpha_1) \to HX(\alpha_2) \to \cdots \to HX(\alpha_n)) \cong \bigoplus_{1 \le b \le d \le n} \mathbb{I}[b, d]^{m_{b,d}},$$

and plot the histogram of its persistence diagram $\{ (\sqrt{\alpha_b}, \sqrt{\alpha_d}) \}$ in $\mathbb{R}^2$.

Here, we present the results carried out over the entire grain packings

- 150,000 spheres with the density 0.60
- 156,315 spheres with the density 0.63
- 216,722 spheres with the density 0.69

in cylindrical containers, as well as 11 non-overlapping subsets each containing 4,000 spheres, which have densities ranging from 0.58 to 0.73. We can see the persistence diagrams in Figures A.2 and A.3. The occurrence rates I_f of the histograms are scaled by the loglog function $f(x) = \log(\log(x+1)+1)$ to highlight fine details of the diagram, and then normalized linearly.

Figure A.2 is a collection of the persistence diagrams of the densities 0.60, 0.63, and 0.69, which are computed from the entire grain packings corresponding to the densities, as well as the persistence diagram of the density 0.73, which is computed from the subset of the grain packing corresponding to the density. Figure A.3 is a collection of the persistence diagrams of 11 subsets.

In both figures, we observe that the points in the persistence diagrams converge to the two points, $(b_t, d_t) = (0.5\sqrt{\frac{1}{3}}, 0.5\sqrt{\frac{1}{2}})$ and $(b_o, d_o) = (0.5\sqrt{\frac{1}{3}}, 0.5)$, as the densities increase. We call those areas around the two points the *tetrahedra spot* and *octahedra spot* respectively, since the two points are generated by the configurations of a regular tetrahedron and regular octahedron (see Figure A.4). A well-known example of the densest sphere packings is face-centered cubic (FCC), whose persistence diagram has only the two points (b_t, d_t) and (b_o, d_o) (with multiplicities). Thus, we are now able to classify the cavities using birth and death value, or its lifetime.

The main advantage using persistent homology is that the persistence diagram is stable to the point cloud, therefore we can define quasi-regular tetrahedral and octahedral cavities, which are difficult to detect for conventional discrete methods (such as Voronoi or Delaunay methods [17]).

A.2. Deformation Models

In the persistence diagrams in Figure A.2, we can observe the populated area, in other words, significant geometric structures. In order to detect such geometric structures, let us consider the following two types of tetrahedral deformation, and two types of octahedral deformation in Figure A.4.

- The first tetrahedral deformation, D1, is that one of the spheres rolls across the saddle area formed by two neighbors, while the other spheres stay fixed. (Figure A.4 **b**)
- The second tetrahedral deformation, D2, is a symmetric process that transforms a regular tetrahedron into a flat square configuration. (Figure A.4 **b**)

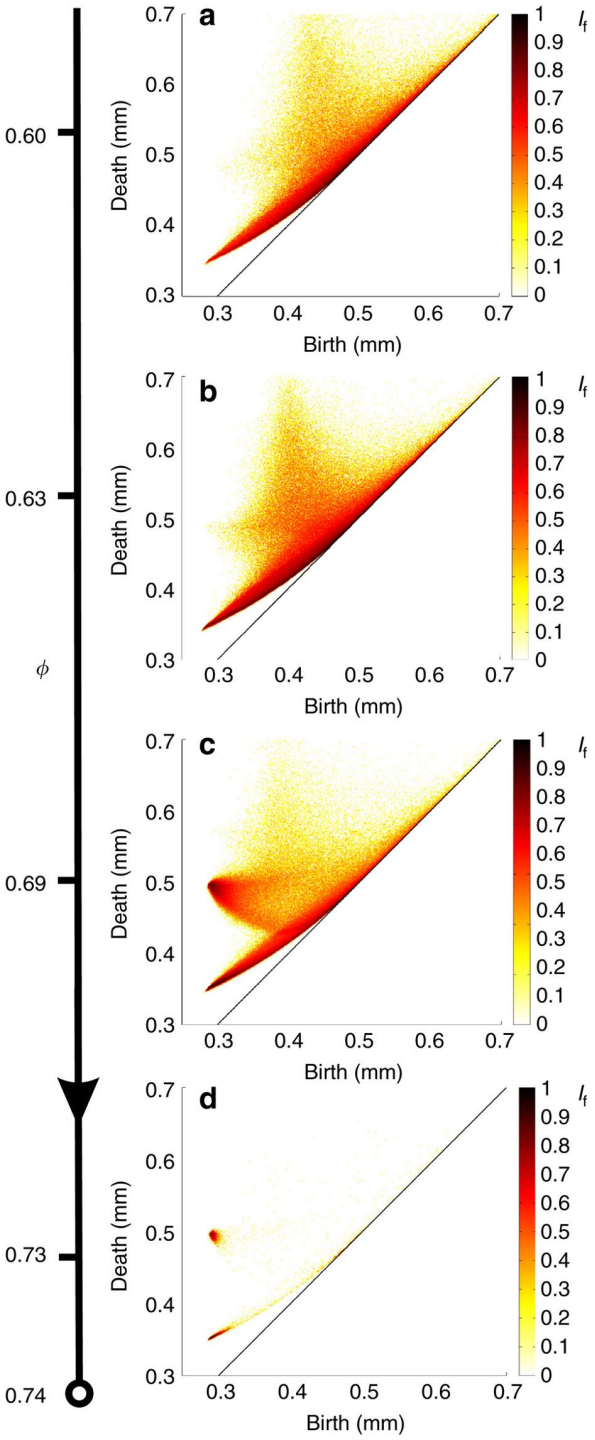


Figure A.2.: Persistence diagrams of sphere packings. **a**, **b**, and **c** are of entire sphere packings, and **d** is of a subset containing 4,000 spheres. (This figure is "PD_2 of disordered, partially ordered and highly ordered sphere packings" by M. Saadatfar et al. https://www.nature.com/articles/ncomms15082/figures/4, used under CC BY 4.0 https://creativecommons.org/licenses/by/4.0/.)

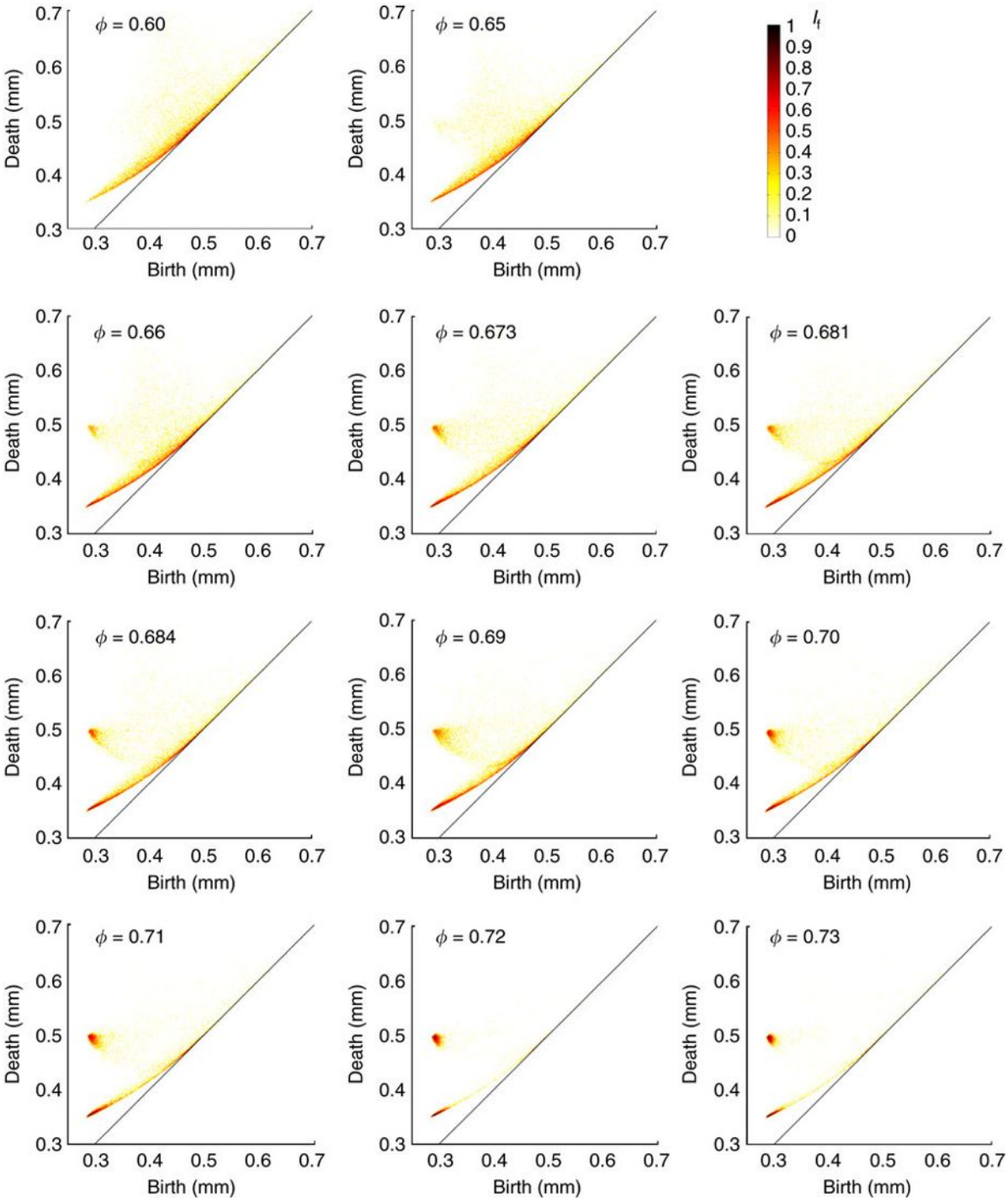


Figure A.3.: Persistence diagrams of 11 subsets each containing 4,000 spheres. The points in persistence diagrams converge to tetrahedral and octahedral spots as the density increases. (This figure is "Evolution of persistence diagram as a function of packing density ϕ" by M. Saadatfar et al. https://www.nature.com/articles/ncomms15082/figures/8, used under CC BY 4.0 https://creativecommons.org/licenses/by/4.0/.)

- The first octahedral deformation, D3, consists of detaching two pairs of spheres symmetrically while keeping the other spheres touching their neighbors, until two edge-adjacent tetrahedra are formed. (Figure A.4 **c**)
- The second octahedral deformation, D4, is analogous to D1. One of the spheres rolls along the saddle area formed by two neighbors, while the other spheres stay fixed. (Figure A.4 **c**)

For each deformation, the persistence diagram changes continuously, that is, each point in the persistence diagram forms a curve during the deformation. Figure A.4 **a** shows these curves for all deformations superimposed on the persistence diagram of the entire packing with the density 0.69.

Since the locations of the curves match well with the boundaries of the populated area, if we assume the hypothesis that the persistence diagram of the partially crystallized packing with the density 0.69 captures some significant structures of the crystallization process, then the four deformation models can be thought of as dominant deformation structures of the crystallization process.

What is problem?

This analysis is a significant result, because persistence diagrams are the first stable method that is able to continuously capture the configurations of cavities thanks to the stability theorem. However, since this analysis focuses only on the persistence diagrams, it does not capture the actual deformation of the inner geometrical structures. To execute the topological analysis on such deformations, we need to study the method which can classify the cavities in deformations. This is a motivation for topological analysis of deformations of point clouds as presented in Chapter 1 and 2-D persistence modules as presented in Subsection 4.4.2.

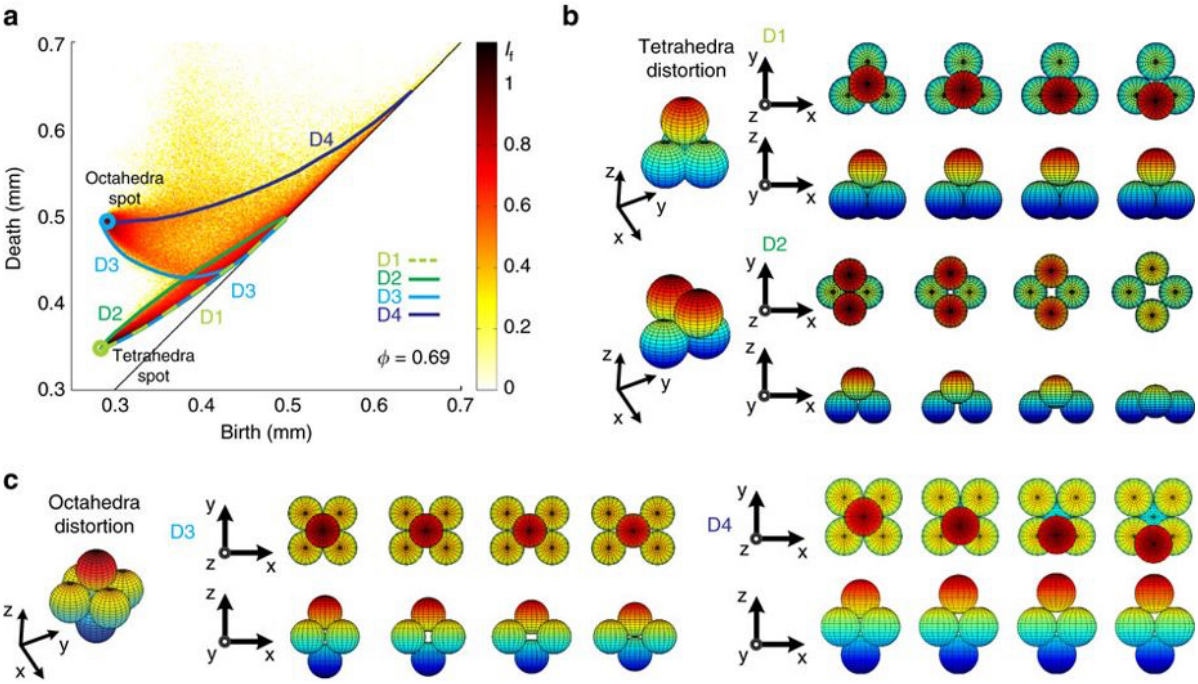


Figure A.4.: Tetrahedral and octahedral deformation scenarios. **a** is the persistence diagram of the entire packing with the density 0.69, with superimposed curves corresponding to the four deformation models. **b** shows tetrahedral deformation models and **c** shows octahedral deformation models. (This figure is "Grain-scale tetrahedral and octahedral formation/deformation scenarios" by M. Saadatfar et al. https://www.nature.com/articles/ncomms15082/figures/6, used under CC BY 4.0 https://creativecommons.org/licenses/by/4.0/, changed by the author.)

Acknowledgements

First of all, I would like to show my greatest appreciation to my supervisor Professor Yasuaki Hiraoka from my master's course to the present, for seminars, discussion, joint works, and innumerable and varied experiences these five years. Without his guidance and persistent help, this thesis would not have been possible. I would like to express my gratitude to my supervisor in the final year of my PhD course, Professor Hiroshi Suito, for his support, helpful advice, and agreeing to be the chair of my thesis committee. I would also like to offer my special thanks to Professor Shigetoshi Bando, who is a vice chair in my thesis committee.

I would particularly like to thank my collaborator, Dr. Emerson G. Escolar, for his massive help in the study of quiver representations and programming, and for wide-ranging discussion. For my other collaborators, I am deeply grateful to Professor Hideto Asashiba, Dr. Mohammad Saadatfar, Dr. Vanessa Robins, and Dr. Nicolas Francois.

I have difficulty in writing English articles. If my English skill has improved, I am indebted to the people around me, particularly the above collaborators and Dr. Killian Meehan.

I would also like to express my gratitude to Japan Society for the Promotion of Science, and Japan Science and Technology Agency for their financial support.

I would like to thank all of the current and past members of Hiraoka laboratory in Kyushu University, Tohoku University, and Kyoto University, for their kindness, support, and encouragement. In addition, I got many inspirations and received sincere encouragement from my friends. Above all, I have to express my special thanks to Professor Ippei Obayashi, Professor Yusuke Imoto, Professor Koya Sakakibara, Professor Yuki Arano, Dr. Kei Hasegawa, Dr. Genki Kusano, Dr. Tomohiro Kanda, and Dr. Yoshiaki Okumura.

Finally, I wish to thank my family for their support and warm encouragement throughout my study.

Bibliography

[1] H. Asashiba, E. G. Escolar, Y. Hiraoka, and H. Takeuchi, Matrix Method for Persistence Modules on Commutative Ladders of Finite Type, Jpn. J. Ind. Appl. Math. DOI:10.1007/s13160-018-0331-y

[2] I. Assem, D. Simson, and A. Skowroński, Elements of the Representation Theory of Associative Algebras 1: Techniques of Representation Theory, Cambridge University Press, Cambridge, 2016.

[3] U. Bauer and M. Lesnick, Induced Matchings of Barcodes and the Algebraic Stability of Persistence, In Proceedings of the Thirtieth Annual Symposium on Computational Geometry (SoCG '14), ACM, New York, NY, USA, 355–364.

[4] K. Borsuk, On the Imbedding of Systems of Compacta in Simplicial Complexes, Fund. Math. 35 (1948), 217–234.

[5] G. Carlsson and V. de Silva, Zigzag Persistence, Found. Comput. Math. 10 (2010), 367–405.

[6] G. Carlsson, V. de Silva, and D. Morozov, Zigzag Persistent Homology and Real-valued Functions, Proc. 25th Ann. Symp. Comput. Geom. ACM (2009).

[7] G. Carlsson and A. Zomorodian, The Theory of Multidimensional Persistence, Discrete Comput. Geom. 42 (2009), 71–93.

[8] F. Chazal, D. Cohen-Steiner, M. Glisse, L. J. Guibas, and S. Y. Oudot, Proximity of Persistence Modules and their Diagrams, In Proceedings of the Twenty-fifth Annual Symposium on Computational Geometry (SoCG '09), ACM, New York, NY, USA, 237–246.

[9] A. Chistov, G. Ivanyos, and M. Karpinski, Polynomial Time Algorithms for Modules over Finite Dimensional Algebras, Proc. 1997 Int. Symp. Symbolic and Algebraic Comput. ACM (1997).

[10] D. Cohen-Steiner, H. Edelsbrunner, and J. Harer, Stability of Persistence Diagrams, Discrete Comput. Geom. 37 (2007), 103–120.

[11] W. Crawley-Boevey, Decomposition of Pointwise Finite-Dimensional Persistence Modules, J. Algebra Appl. 14 (2015), 1550066.

[12] H. Edelsbrunner, The Union of Balls and Its Dual Shape, Discrete Comput. Geom. 13 (1995), 415–440.

[13] H. Edelsbrunner and J. Harer, Computational Topology: An Introduction, Amer. Math. Soc., Providence, Rhode Island, 2010.

[14] H. Edelsbrunner, G. Jabłoński, and M. Mrozek, The Persistent Homology of a Self-Map, Found. Comput. Math. 15 (2015), 1213–1244.

[15] H. Edelsbrunner, D. Letscher, and A. Zomorodian, Topological Persistence and Simplification, Discrete Comput. Geom. 28 (2002), 511–533.

[16] E. G. Escolar and Y. Hiraoka, Persistence Modules on Commutative Ladders of Finite Type, Discrete Comput. Geom. 55 (2016), 100–157.

[17] N. Francois, M. Saadatfar, R. Cruikshank, and A. Sheppard, Geometrical Frustration in Amorphous and Partially Crystallized Packings of Spheres, Phys. Rev. Lett. 111 (2013), 148001.

[18] P. Gabriel, Unzerlegbare Darstellungen I, Manuscripta Math. 6 (1972), 71–103.

[19] M. Gameiro, Y. Hiraoka, S. Izumi, M. Kramar, K. Mischaikow, and V. Nanda, Topological Measurement of Protein Compressibility via Persistent Diagrams, Jpn. J. Ind. Appl. Math. 32 (2015), 1–17.

[20] E. L. Green, L. S. Heath, and C. A. Struble, Constructing Homomorphism Spaces and Endomorphism Rings, J. Symb. Comput. 32 (2001), 101–117.

[21] S. Harker, H. Kokubu, K. Mischaikow, and P. Pilarczyk, Inducing a Map on Homology from a Correspondence, Proc. Amer. Math. Soc. 144 (2016), 1787–1801.

[22] Y. Hiraoka, T. Nakamura, A. Hirata, E. G. Escolar, K. Matsue, and Y. Nishiura. Hierarchical Structures of Amorphous Solids Characterized by Persistent Homology. Proc. Natl. Acad. Sci. USA 113 (2016), 7035–7040.

[23] M. Lesnick, The Theory of the Interleaving Distance on Multidimensional Persistence Modules, Found. Comput. Math., 15 (2015), 613–650.

[24] K. M. Lux and M. Szőke, Computing Decompositions of Modules over Finite-dimensional Algebras, Experimental Math. 16(1) (2007), 1–6.

[25] K. M. Lux and M. Szőke, Computing Homomorphism Spaces between Modules over Finite Dimensional Algebras, Experimental Math. 12(1) (2003), 91–98.

[26] M. Saadatfar, H. Takeuchi, V. Robins, N. Francois, and Y. Hiraoka, Pore Configuration Landscape of Granular Crystallization, Nat. Commun. 8 (2017), 15082.

[27] H. Takeuchi, The Persistent Homology of a Sampled Map: From a Viewpoint of Quiver Representations, arXiv:1810.11774 (2018).

[28] A. Zomorodian and G. Carlsson, Computing Persistent Homology. Discrete Comput. Geom. 33(2) (2005), 249–274.

[29] CGAL, Computational Geometry Algorithms Library. https://www.cgal.org

[30] G. Guennebaud, B. Jacob, and others, Eigen v3 (2010). http://eigen.tuxfamily.org